

MARISA JOÃO GUERRA PEREIRA DE OLIVEIRA

Modelos Matemáticos e Heurísticas para Problemas de Posicionamento de Polígonos Ortogonais

Tese apresentada à Faculdade de Engenharia da Universidade do Porto
para obtenção do grau de Doutor em Engenharia Industrial e Gestão sob
orientação científica de António Miguel da Fonseca Fernandes Gomes e de
Maria Eduarda da Cunha e Silva Pinto Ferreira

DEPARTAMENTO DE ENGENHARIA E GESTÃO INDUSTRIAL
FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO
2012

Agradecimentos

Gostaria de agradecer, em primeiro lugar, ao Prof. Miguel Gomes, meu orientador científico, sem o qual não teria conseguido realizar este trabalho. Exprimo-lhe o meu reconhecimento por todos os ensinamentos que me transmitiu. Não há palavras suficientes para traduzir tal agradecimento.

Agradeço à minha Co-Orientadora, Doutora Eduarda Pinto Ferreira, por toda a sua disponibilidade para ajudar e pelas suas palavras de apoio ao longo deste percurso.

Agradeço também ao Prof. José Fernando Oliveira pelas suas sugestões, apoio e palavras de incentivo tão importantes nos momentos menos bons.

Agradeço à Teresa Bianchi de Aguiar e ao Luís Guimarães pela enorme ajuda com o Visual Studio e com o Cplex.

Não poderia deixar de agradecer a ajuda prestada, nas atividades letivas, pela minha colega e amiga Graça Marcos.

Gostaria de agradecer ao ISEP e ao Programa de Formação Avançada de Docentes do IPP, pelo apoio financeiro parcial.

E, finalmente, agradeço aos meus pais por todo o apoio, amor e carinho e porque sempre me incentivaram à formação.

Resumo

O Problema de Posicionamento abordado neste trabalho consiste em minimizar a área da envolvente rectangular que contém um conjunto de itens ortogonais (i.e. itens cujos ângulos internos sejam 90^0 ou 270^0). Os problemas de posicionamento são problemas de otimização combinatória com uma forte componente geométrica e são, na sua generalidade, problemas NP-Difíceis do ponto de vista da sua resolução. O problema abordado neste trabalho surge em vários contextos reais: tais como no posicionamento de módulos em circuitos VLSI, que tem como objetivo encontrar o melhor posicionamento dos módulos para obter circuitos compactos, e no projeto físico de instalações, no qual um dos objetivos passa por encontrar o arranjo mais eficiente de departamentos dentro de uma instalação. Por exemplo, no posicionamento de itens, em circuitos VLSI módulos com formas ortogonais surgiram para facilitar a utilização da área do circuito e melhorar a ligação entre os itens, contribuindo para o aumento do desempenho do circuito. As instâncias que surgem nos problemas reais, principalmente nos circuitos integrados, apresentam um elevado número de itens que podem mesmo chegar aos milhares, sendo um grande desafio lidar com instâncias desta dimensão. Enquanto problemas de otimização combinatória podem ser resolvidos usando qualquer uma das abordagens e técnicas disponíveis: programação matemática, heurísticas, meta-heurísticas, algoritmos de pesquisa, programação por restrições, etc. Apesar de serem facilmente representados através de modelos matemáticos são difíceis de serem solucionados dado o grande número de variáveis envolvidas e, consequentemente, obter soluções exatas, principalmente quando aplicados a instâncias de grande dimensão. Desta forma, são muitas vezes usados métodos heurísticos para os solucionar. Apesar destes métodos não garantirem a obtenção da solução ótima do problema, podem obter boas aproximações, com um baixo esforço computacional, quando comparados com a utilização de métodos exatos.

Abstract

In the rectilinear packing area minimization problem one wishes to pack a set of rectilinear shaped pieces (pieces with 90^0 or 270^0 internal angles), while minimizing the area of the enclosing rectangle and without overlapped pieces. This is a combinatorial optimization NP-hard problem that belongs to the class of Cutting and Packing problems, which are combinatorial problems with a strong geometric component.

Packing problems occur in several important manufacturing industries where the reduction of raw materials leads to environmental and ecological benefits. In most of the industrial applications the goals are similar: to produce fast and good quality solutions in order to maximize material utilization and, therefore, minimize wastage. The rectilinear packing problem arises in a wide range of activities, such as in the placement of circuit modules in Very Large Scale Integration (VLSI) circuits, in the design of facility layouts, newspaper layout, etc. For example, in VLSI circuits rectilinear shaped pieces appeared to reduce the circuits' area while improving the connectivity between pieces and increasing the circuit performance. In this real world application it is important to meet some market requirements such as the quick production and design of small circuits. VLSI circuits today may contain a high number of pieces which may be rectangles or rectilinear shapes. For these reasons is important to minimize the area of the enclosing rectangle that contains all the circuit elements placed in order to obtain small chips. It is also important to have an approach which deals with a large number and variety of pieces, where, besides rectangular shapes, it could deal with rectilinear shaped pieces, and which produces fast and good quality solutions. Several techniques are used to solve the rectilinear packing problems, like mixed integer programming models, heuristics, metaheuristics, etc. Exact techniques guarantee to find optimal solutions and prove its optimality, however given the combinatorial nature of rectili-

near packing problems they are not able to deal with large instances. The most common approaches to tackle large instances are based on heuristic methods. These methods are designed to obtain good solutions, not necessarily optimal, in acceptable computational time. In this work we intend to solve large rectilinear packing problems in a reasonable computational time.

Conteúdo

| | | |
|----------|---|-----------|
| 1 | Introdução | 1 |
| 1.1 | Motivação e âmbito da Tese | 1 |
| 1.2 | Objetivos | 2 |
| 1.3 | Estrutura da Tese | 3 |
| 2 | Problemas de Posicionamento de Polígonos Ortogonais | 5 |
| 2.1 | Problemas de Posicionamento | 7 |
| 2.1.1 | Tipologia de Problemas de Posicionamento | 8 |
| 2.1.2 | Geometria em Problemas de Posicionamento | 22 |
| 2.2 | Posicionamento de Retângulos em Objetos Definidos | 27 |
| 2.3 | Problemas de Minimização da Área do Objeto | 32 |
| 2.3.1 | Problemas com Itens Retangulares | 32 |
| 2.3.2 | Problemas com Itens Ortogonais | 39 |
| 2.4 | Posicionamento Multiobjetivo de Retângulos | 43 |
| 2.4.1 | Posicionamento de Módulos em Circuitos VLSI | 43 |
| 2.4.2 | Projeto Físico de Instalações | 45 |
| 2.4.3 | Outros Problemas de Posicionamento Multiobjetivo | 46 |
| 3 | Modelos para Posicionamento de Polígonos Ortogonais | 51 |
| 3.1 | Restrições de Não Sobreposição | 53 |
| 3.2 | Posicionamento em Objetos Definidos | 59 |
| 3.3 | Problemas de Minimização da Área do Objeto | 62 |
| 3.3.1 | Aproximações Lineares ao Cálculo da Área | 64 |
| 3.3.2 | Aproximações ao Cálculo da Área baseadas em Funções Definidas por Ramos | 66 |
| 3.4 | Modelo Matemático Desenvolvido | 70 |
| 3.5 | Problema Integrado de Minimização da Área e Distâncias | 73 |
| 3.5.1 | Aproximações ao Cálculo de Distâncias | 76 |

| | | |
|----------|---|------------|
| 3.5.2 | Aproximações à Minimização da Área e Distâncias . . . | 80 |
| 3.6 | Experiências Computacionais | 83 |
| 4 | Heurísticas de Posicionamento de Minimização da Área | 91 |
| 4.1 | Heurística de Construção de Padrões Guilhotináveis | 92 |
| 4.1.1 | Representação de Padrões Guilhotináveis | 92 |
| 4.1.2 | Heurística FAST-GR | 95 |
| 4.1.3 | Testes Computacionais | 96 |
| 4.1.4 | Comentários e Conclusões | 102 |
| 4.2 | Heurística para Padrões Não Guilhotináveis | 110 |
| 4.2.1 | Invólucro de Posicionamento | 110 |
| 4.2.2 | Heurística Rect-TOPOS | 112 |
| 4.2.3 | Testes Computacionais | 115 |
| 4.2.4 | Comentários e Conclusões | 121 |
| 5 | Abordagens Híbridas para Probl. de Minimização da Área | 123 |
| 5.1 | Abordagem Híbrida ILS+Compactação/Separação | 126 |
| 5.1.1 | Meta-Heurística ILS | 127 |
| 5.1.2 | Modelos de Compactação e Separação | 128 |
| 5.1.3 | Testes Computacionais | 130 |
| 5.1.4 | Comentários e Conclusões | 132 |
| 5.2 | Abordagem Híbrida Baseada na Construção de Padrões . . . | 134 |
| 5.2.1 | Heurística Construtiva Híbrida | 135 |
| 5.2.2 | Análise às Instâncias | 142 |
| 5.2.3 | Testes Computacionais | 144 |
| 5.2.4 | Comentários e Conclusões | 151 |
| 6 | Conclusão | 153 |
| 6.1 | Conclusões do Trabalho | 153 |
| 6.2 | Desenvolvimentos Futuros | 154 |
| | Bibliografia | 157 |

Lista de Figuras

| | | |
|------|--|----|
| 2.1 | Problema de posicionamento bidimensional | 6 |
| 2.2 | Posicionamento de polígonos ortogonais | 7 |
| 2.3 | Categorias de Problemas de Posicionamento (adaptado de Wäscher et al. [2007].) | 11 |
| 2.4 | Problemas de posicionamento (adaptado de Oliveira et al. [2011a]) | 13 |
| 2.5 | Polígonos regulares e não regulares, convexos e não convexos | 24 |
| 2.6 | Padrão guilhotinável e não guilhotinável | 25 |
| 2.7 | Exemplo de um padrão não ortogonal | 25 |
| 2.8 | Exemplo de um padrão periódico (adaptado de Costa et al. [2009]) | 25 |
| 2.9 | Itens com orientação fixa e orientação de x graus | 26 |
| 2.10 | Árvore que representa um padrão guilhotinável e a correspondente <i>Normalized Polish Expression</i> | 34 |
| 2.11 | Grafos de restrição. | 35 |
| 2.12 | Representação SP (adaptado de Murata et al. [1995]) | 35 |
| 2.13 | Representação BSG (adaptado de Nakatake e Fujiyoshi [1997]) | 36 |
| 2.14 | O- <i>tree</i> (adaptado de Yoshimura [1999]). | 37 |
| 2.15 | B*- <i>tree</i> (adaptado de Chang et al. [2000]). | 37 |
| 2.16 | Posicionamento e correspondentes TCG (horizontal e vertical) (adaptado de Lin e Chang [2001]). | 38 |
| 2.17 | Circuito VLSI | 44 |
| 2.18 | Maximização do aproveitamento e minimização do custo de corte (adaptado de Moreira [2001]) | 48 |
| 3.1 | Valores das variáveis binárias r_{ij} e u_{ij} em função do posicionamento relativo entre dois itens i e j | 54 |

| | | |
|------|---|-----|
| 3.2 | Relação entre as variáveis binárias r_{ij} e u_{ij} em função do posicionamento relativo entre dois itens i e j | 55 |
| 3.3 | Valores das variáveis binárias r_{ij} e u_{ij} para uma situação de sobreposição entre dois itens i e j | 56 |
| 3.4 | Divisão de um polígono ortogonal em retângulos | 57 |
| 3.5 | Rotações possíveis de itens ortogonais | 60 |
| 3.6 | Aproximação por ramos (adaptado de Li et al. [2002]) | 70 |
| 3.7 | Distância de Manhattan = $F_x + F_y$ | 77 |
| 3.8 | Diferentes formas de calcular as distâncias | 79 |
| 3.9 | Influência do desperdício na minimização das distâncias (adaptado de Tang et al. [2006]) | 80 |
| 3.10 | Exemplo de um padrão com 7 retângulos | 84 |
| 3.11 | Exemplo de um padrão com 4 itens ortogonais e 1 retângulo | 86 |
| 3.12 | Fronteira eficiente para a minimização da área e da soma das distâncias | 89 |
| 4.1 | Heurística Construtiva FAST-GR | 93 |
| 4.2 | Operações usadas na união de dois retângulos | 94 |
| 4.3 | Posição relativa de dois retângulos no interior do retângulo exterior envolvente | 94 |
| 4.4 | Seleção dos itens usando o parâmetro N_1 e o parâmetro N_2 | 95 |
| 4.5 | Fluxograma da heurística construtiva FAST-GR | 96 |
| 4.6 | Variação do valor da função objetivo para cada classe considerada | 99 |
| 4.7 | Variação do valor da função objetivo com o número total de retângulos | 99 |
| 4.8 | Variação do valor da função objetivo com o parâmetro N_1 | 100 |
| 4.9 | Variação do valor da função objetivo com o parâmetro N_2 | 101 |
| 4.10 | Variação do valor da função objetivo para N_1 considerando N_2 igual a <área, <perímetro e <desperdício | 101 |
| 4.11 | Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continua) | 103 |
| 4.12 | Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continuação e continua) | 104 |
| 4.13 | Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continuação e continua) | 105 |

| | | |
|------|--|-----|
| 4.14 | Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continuação) | 106 |
| 4.15 | Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continua) | 107 |
| 4.16 | Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continuação) | 108 |
| 4.17 | Invólucro de posicionamento do item j em relação ao item i (NFP_{ij}) | 111 |
| 4.18 | Construção da solução parcial Oliveira et al. [2000] | 113 |
| 4.19 | Crerios usados na escolha do próximo item a posicionar | 114 |
| 4.20 | Crerrio da sobreposição entre retângulos | 114 |
| 4.21 | Construção de itens ortogonais nas instâncias geradas | 116 |
| 4.22 | Solução obtida para a instância PCB500 | 120 |
| 5.1 | Classificação resultante da combinação entre modelos matemáticos e meta-heurísticas (adaptado de Puchinger e Raidl [2005]) | 125 |
| 5.2 | Hibridização entre modelos matemáticos e meta heurísticas | 126 |
| 5.3 | Estrutura da abordagem híbrida ILS+Compactação/Separação | 127 |
| 5.4 | Estrutura detalhada da abordagem híbrida ILS+Compactação/Separação | 129 |
| 5.5 | Otimização local | 130 |
| 5.6 | Compactação e separação de itens | 131 |
| 5.7 | Exemplo de uma operação de troca de posições entre dois itens e da aplicação dos modelos de compactação e separação | 131 |
| 5.8 | Exemplos de padrões obtidos através da abordagem ILS+Compactação/Separação | 134 |
| 5.9 | Ilustração do funcionamento da heurística construtiva | 137 |
| 5.10 | Fluxograma da heurística construtiva hibridizada com modelos matemáticos | 139 |
| 5.11 | Variação da área dos itens em cada instância | 143 |
| 5.12 | Padrões obtidos para as instâncias retangulares com a heurística construtiva híbrida numa configuração mais lenta | 147 |
| 5.13 | Padrões obtidos para as instâncias retangulares com a heurística construtiva híbrida numa configuração mais rápida | 148 |
| 5.14 | Padrões obtidos para as instâncias LIN e NAKATAKE1 | 152 |

Lista de Tabelas

| | | |
|------|---|-----|
| 3.1 | Variáveis de decisão e auxiliares | 53 |
| 3.2 | Parâmetros | 53 |
| 3.3 | Parâmetros relacionados com polígonos ortogonais | 57 |
| 3.4 | Variáveis de decisão e auxiliares relacionadas com polígonos ortogonais | 58 |
| 3.5 | Variáveis de decisão auxiliares para a modelação da área . . . | 63 |
| 3.6 | Variáveis de decisão e auxiliares (modelo completo) | 71 |
| 3.7 | Parâmetros (modelo completo) | 71 |
| 3.8 | Variáveis de decisão relacionadas com as distâncias | 75 |
| 3.9 | Resultados das experiências computacionais com o modelo matemático de minimização da área do objeto para itens re- tangulares | 84 |
| 3.10 | Resultados das experiências computacionais com o modelo matemático de minimização da área do objeto para itens or- togonais | 85 |
| 3.11 | Fronteira eficiente do problema integrado de minimização da área e das distâncias | 87 |
| 4.1 | Possibilidades de escolha para valores de N_1 e N_2 | 96 |
| 4.2 | Tabela resumo das combinações N_1 e N_2 | 109 |
| 4.3 | Características das instâncias usadas. | 117 |
| 4.4 | Comparação dos resultados obtidos pela heurística Rect-TOPOS vs resultados da literatura. | 118 |
| 4.5 | Comparação dos tempos de execução entre a heurística Rect- TOPOS e a abordagem apresentada em Bortfeldt [2013]. . . | 120 |
| 5.1 | Resultados obtidos pela abordagem híbrida (20 corridas) . . . | 133 |
| 5.2 | Análise estatística e valor do Δ | 144 |

| | | |
|-----|--|-----|
| 5.3 | Melhores resultados conhecidos | 145 |
| 5.4 | Resultados dos testes computacionais da heurística construtiva híbrida numa configuração mais demorada | 146 |
| 5.5 | Resultados dos testes computacionais da heurística construtiva híbrida numa configuração mais rápida | 149 |
| 5.6 | Itens ortogonais das instâncias NAKATAKE1 e LIN | 150 |
| 5.7 | Resultados dos testes computacionais de instâncias com itens ortogonais | 150 |

Capítulo 1

Introdução

1.1 Motivação e âmbito da Tese

Um problema de posicionamento (PP) consiste em posicionar um conjunto de itens, sem sobreposições, com o objetivo de minimizar o desperdício, isto é, pretende-se minimizar a área não ocupada pelos itens. Os PP são problemas de Otimização Combinatória que estão associados a duas componentes principais, uma geométrica que resulta da necessidade de posicionar os itens sem sobreposições e outra combinatória que resulta de combinar os itens de forma a minimizar o desperdício. Os PP são também conhecidos como problemas de Corte e Empacotamento (*Cutting and Packing*). Exemplos de aplicação incluem a minimização da perda de um material (para corte) ou maximização da área aproveitada de um contentor (para empacotamento). Devido à sua natureza genérica, os PP são utilizados para modelar diversas situações reais que surgem em várias áreas industriais, tais como indústrias de papel, mobiliário, vidro, metalúrgica, plástica, têxtil, passando pela minimização da área utilizada por um conjunto de módulos num circuito VLSI (*Very Large Scale Integration*). São problemas que ocorrem frequentemente relacionados com o desperdício de materiais, planeamento de operações logísticas como armazenagem, movimentação e transporte, minimização de espaços vazios, entre outros. Os PP enquanto problemas de otimização combinatória podem ser resolvidos usando qualquer uma das abordagens e técnicas disponíveis: programação matemática, heurísticas, meta-heurísticas, algoritmos de pesquisa, etc. Infelizmente, dada a sua natureza combinatória e a sua componente geométrica, as técnicas exatas não são capazes de lidar eficientemente com instâncias de grande dimensão destes problemas, pelo

que a utilização de métodos heurísticos se torna indispensável.

O foco principal desta dissertação serão os PP de itens ortogonais com o objetivo de minimizar a área do objeto que os contém.

1.2 Objetivos

O objetivo do PP de polígonos ortogonais, que é alvo de estudo nesta dissertação, consiste em minimizar a área da envolvente retangular que contém um conjunto de itens ortogonais (i.e. itens cujos ângulos internos são 90^0 ou 270^0). São exemplos de itens ortogonais formas em L, T, U. Este problema surge em vários contextos reais, tais como: no posicionamento de módulos em circuitos VLSI, que tem como objetivo encontrar o melhor posicionamento dos módulos para obter circuitos compactos, e no projeto físico de instalações, no qual um dos objetivos passa por encontrar o arranjo mais eficiente de departamentos dentro de uma instalação. Por exemplo, no posicionamento de itens em circuitos VLSI, módulos com formas ortogonais surgiram para facilitar a utilização da área do circuito e melhorar a ligação entre os itens, contribuindo para o aumento da performance do circuito. As instâncias que surgem nos problemas reais, principalmente nos circuitos integrados, apresentam um elevado número de itens que podem mesmo chegar aos milhares, sendo um grande desafio lidar com instâncias desta dimensão.

Várias questões de investigação foram levantadas no início deste trabalho, algumas relacionadas com as abordagens a usar para lidar com o problema proposto e outras relacionadas com a representação dos itens ortogonais.

Em relação às abordagens é necessário decidir qual a abordagem a usar. Será que para este problema existe alguma abordagem que é mais indicada do que outras? Foi inicialmente efetuado um estudo sobre a possibilidade de usar modelos matemáticos para resolver o problema proposto. Pretendendo-se saber até que número de itens é possível resolver o PP até à otimalidade. Quando se lida com modelos matemáticos não-lineares e se pretende usar técnicas de programação linear e inteira é necessário decidir que linearização usar e qual é a que mais se adequa ao problema. Como nos problemas reais surgem frequentemente objetivos adicionais, para além da minimização da área do objeto, foi também estudada a componente multiobjetivo do PP de itens ortogonais quando se pretende minimizar simultaneamente a área

do retângulo envolvente e a soma das distâncias que ligam um conjunto de itens. Aqui, é também necessário decidir como combinar estes dois objetivos e como linearizar a soma das distâncias entre itens.

Para além do recurso a modelos matemáticos foram também usadas abordagens heurísticas, quer métodos construtivos quer métodos baseados em pesquisa local. Nos métodos construtivos é necessário decidir que regra usar na ordenação dos itens para posteriormente se proceder ao seu posicionamento. Vários autores referem que deve ser dada preferência aos itens grandes para serem posicionados em primeiro lugar. Para confirmar esta afirmação foi feito um estudo, recorrendo a uma heurística construtiva parametrizável, que permitiu eliminar alguns critérios e identificar as melhores combinações de parâmetros para a seleção dos itens a posicionar.

Tentando tirar partido das vantagens existentes nos modelos matemáticos e nos métodos heurísticos foram propostas abordagens que combinam estes dois métodos de resolução. Uma das abordagens combina a meta-heurística *Iterated Local Search* (ILS) com modelos matemáticos de compactação e separação. Uma segunda abordagem utiliza modelos matemáticos na construção iterativa de soluções. Em cada iteração um conjunto de itens é adicionado a uma solução parcial, por intermédio de um modelo matemático. A escolha dos itens a posicionar em cada iteração é feita segundo critérios heurísticos.

Outra questão que se levanta está relacionada com os itens ortogonais. Nas abordagens usadas é mais fácil lidar com estes itens como um todo ou devem ser divididos em retângulos e impor um conjunto de restrições adicionais que assegurem que na solução final a forma inicial dos itens ortogonais é mantida? Em algumas das abordagens propostas nesta dissertação optou-se pela divisão dos itens ortogonais em retângulos e noutras optou-se por lidar com o item como um todo.

1.3 Estrutura da Tese

Esta tese divide-se em seis capítulos, sendo o presente capítulo uma introdução do problema que se pretende abordar, apresentando-se as principais motivações, questões de investigação e objetivos.

No segundo capítulo é feito um enquadramento do problema objeto de estudo desta dissertação. Começa-se por enquadrá-lo nos PP com base na

mais recente tipologia existente para a classificação destes problemas. É apresentada uma referência a publicações existentes na área de PP bidimensionais que envolvem itens retangulares e polígonos ortogonais.

No terceiro capítulo são apresentados modelos matemáticos para o PP de minimização da área do objeto com itens ortogonais. São realizados três conjuntos de experiências com objetivos diferentes. Um dos conjuntos visa estabelecer as capacidades do modelo proposto em relação ao número de itens considerados. Outro tem como objetivo principal validar a extensão do modelo para lidar com itens ortogonais e comparar a complexidade dos modelos que lidam com itens ortogonais com modelos que consideram apenas itens retangulares. No último conjunto de experiências pretendeu-se estudar se existe uma componente multiobjetivo quando se considera também o objetivo de minimizar a soma das distâncias entre itens.

No quarto capítulo é proposta uma heurística construtiva parametrizável que usa uma estrutura em árvore para representar padrões guilhotináveis. Com esta heurística pretende-se fazer um estudo de quais os itens que devem ser posicionados em primeiro lugar. Neste capítulo apresenta-se ainda uma heurística, também construtiva, cuja solução são padrões não guilhotináveis.

O quinto capítulo é dedicado à apresentação de abordagens híbridas. Estas abordagens combinam modelos matemáticos com métodos heurísticos.

E, finalmente, no sexto capítulo faz-se uma síntese dos principais assuntos abordados nesta tese, apresentam-se as principais conclusões e são apresentadas algumas propostas de trabalho futuro.

Capítulo 2

Problemas de Posicionamento de Polígonos Ortogonais

Os PP são problemas de otimização combinatória com uma forte componente geométrica. Estes problemas também são designados por problemas de Corte e Empacotamento (*Cutting and Packing Problems*, na versão inglesa). A componente geométrica dos PP está presente no posicionamento dos itens, uma vez que é necessário assegurar que estes não se sobrepõem, enquanto a componente combinatória surge na seleção dos itens, no posicionamento, na seleção do objeto grande, etc. O objetivo nestes problemas poderá ser a maximização do valor dos objetos menores (designados por itens) a serem posicionados num objeto grande (designado apenas por objeto) ou a minimização de objetos para posicionar os itens, figura 2.1. As operações de corte ou de empacotamento são frequentes em diversas indústrias como as de madeira, têxtil, vidro, aço, espuma, construção civil, etc. Nestas indústrias, um importante objetivo passa por minimizar o desperdício da matéria prima. O processo de cortar objetos em itens ou empacotar itens em objetos são problemas equivalentes, matematicamente falando, uma vez que apresentam a mesma estrutura lógica podendo ser abordados com as mesmas formulações e estratégias para a obtenção da solução.

Neste trabalho, abordamos o problema de posicionamento de polígonos ortogonais em que se pretende posicionar um conjunto de itens, sem sobreposições, de forma a otimizar um ou mais objetivos. Na Figura 2.2 representa-

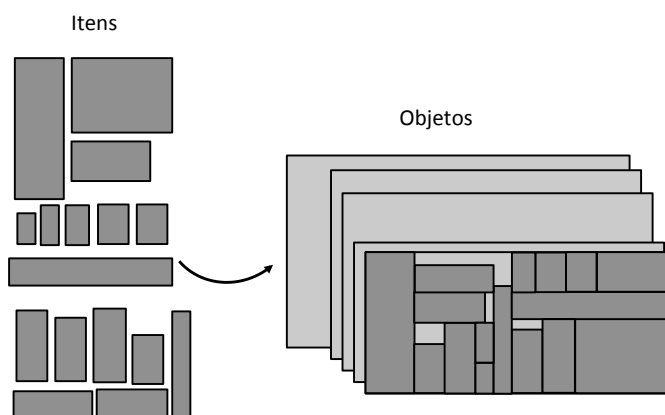


Figura 2.1: Problema de posicionamento bidimensional

se uma solução de um PP que contém polígonos ortogonais em que o objetivo é minimizar a área do retângulo exterior envolvente que contém todos os itens disponíveis para o posicionamento, sem que estes se sobreponham. Para além da minimização da área, considerar-se-á um outro objetivo, que surge frequentemente em aplicações reais nomeadamente no posicionamento de módulos em circuitos, onde se pretende minimizar a soma das distâncias entre um determinado subconjunto de itens.

Este capítulo divide-se em três secções. A primeira secção aborda os PP, fazendo uma referência à tipologia mais recente usada na classificação destes problemas. São também referidas as categorias e subcategorias assim como alguns trabalhos e abordagens usadas nestes problemas. Nesta secção será realizada uma descrição da geometria e das restrições adicionais existentes nos PP.

Na secção seguinte será realizada uma breve descrição das principais abordagens a PP com itens retangulares a serem posicionados num conjunto de objetos com dimensões conhecidas. Apesar dos problemas apresentados, nesta secção, serem diferentes do problema que é o tema de estudo desta dissertação, existem algumas características que são comuns, podendo as abordagens usadas para estes problemas serem usadas para o problema em estudo, através de algumas adaptações. Esta secção começa por uma breve contextualização histórica de como surgiram estes problemas, seguindo-se a apresentação de um conjunto de artigos seleccionados para cada problema.

A terceira secção é dedicada à apresentação do problema que é o tema de estudo desta dissertação. Neste problema, um conjunto de itens tem

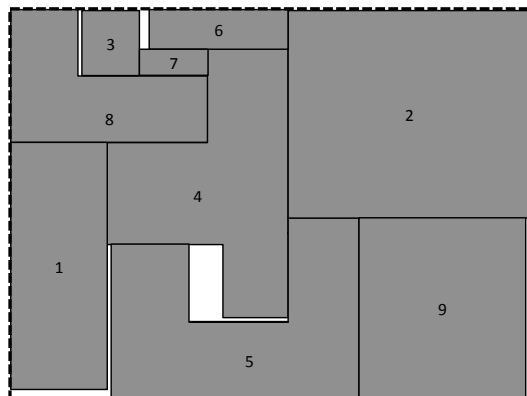


Figura 2.2: Posicionamento de polígonos ortogonais

que ser posicionado sem sobreposições. O objeto para posicionar os itens não tem qualquer dimensão definida sendo, o objetivo minimizar a área do objeto que contém todos os itens. Serão apresentados algumas aplicações reais onde surge este problema, assim como uma revisão bibliográfica das abordagens usadas. Esta revisão bibliográfica está dividida em duas partes, uma que considera apenas itens retangulares e outra onde os itens podem ser polígonos ortogonais.

Por último, realiza-se uma breve revisão bibliográfica de PP multiobjetivo (PPMO). Na primeira parte desta secção faz-se referência a problemas e abordagens onde, para além da minimização da área, é considerado um objetivo adicional. A segunda parte é dedicada a outros PPMO, por exemplo, problemas onde os itens ou os objetos têm formas irregulares ou problemas que são extensões de PP.

2.1 Problemas de Posicionamento

Os PP são, na sua generalidade, problemas NP-Difíceis do ponto de vista da sua resolução (Garey e Johnson [1979]). Korf et al. [2008] mostram que o PP retangular é NP-completo, através de uma redução de *Bin Packing*. Enquanto problemas de otimização combinatória podem ser resolvidos usando qualquer uma das abordagens e técnicas disponíveis: programação linear, heurísticas, meta-heurísticas, algoritmos de pesquisa, programação por restrições, etc. Apesar de serem facilmente representados através de modelos matemáticos são difíceis de serem solucionados dado o grande número de

variáveis envolvidas e, conseqüentemente, obter soluções exatas em tempos computacionais razoáveis, principalmente quando aplicados a instâncias de grande dimensão. Desta forma, são muitas vezes usados métodos heurísticos para os solucionar. Apesar destes métodos não garantirem a obtenção da solução ótima do problema, podem obter boas aproximações, com um baixo esforço computacional, quando comparados com a utilização de métodos exatos.

2.1.1 Tipologia de Problemas de Posicionamento

Nas últimas décadas foram publicados vários artigos relacionados com PP. Dada a enorme variedade de problemas que surgiram nesta área tornou-se necessário a sua sistematização para facilitar a identificação da estrutura comum existente nestes problemas. Apresenta-se de seguida a tipologia, mais recente, usada na literatura para classificar os PP com o objetivo de ilustrar melhor os vários contextos em que surgem estes problemas. Uma tipologia tem por objetivo facilitar a classificação das variações existentes em problemas de uma determinada área, permitindo a normalização de conceitos, definições e notações e, assim, facilitar a comunicação entre investigadores desta área.

Na literatura, podemos encontrar duas formas de classificação dos PP: a descrita por Dyckhoff [1990], e a de Wäscher et al. [2007]. A tipologia proposta por Dyckhoff [1990], apesar de ter sido um marco importante no estabelecimento de uma classificação que destaca a estrutura básica comum aos PP, não obteve uma unanimidade internacional. O seu mecanismo de codificação foi considerado pouco evidente e inconsistente, cuja aplicação pode levar a resultados confusos, além de se encontrar em alemão. Para além de que, ao longo dos anos, esta tipologia não se revelou eficaz para caracterizar novos problemas que foram surgindo. Mais tarde, surgiu uma nova tipologia proposta por Wäscher et al. [2007] onde a codificação apresentada para a classificação destes problemas é clara e acrescenta melhorias à tipologia anteriormente existente. Esta tipologia ampliou a classificação proposta incluindo uma maior quantidade de problemas.

Segundo Wäscher et al. [2007], podem ser distinguidos cinco subproblemas de PP nomeadamente: problema de seleção de objetos, que surge quando os objetos possuem características diferentes (como dimensões, custo, material); problema de seleção de itens, que ocorre quando determinados

itens têm prioridade em relação aos outros; problema de agrupamento de itens, onde um determinado conjunto de itens não pode ficar junto de outro, como, por exemplo, produtos inflamáveis com produtos alimentícios, itens frágeis com outros itens que os possam danificar; problema de alocação de itens em objetos quando determinados itens só podem ser posicionados em determinados objetos; e, finalmente, problema de padrão que surge quando os itens devem ser dispostos nos objetos, respeitando-se determinadas condições geométricas, por exemplo, quando os itens têm posições fixas no interior de um objeto.

Para os diferentes problemas, segundo esta tipologia, há cinco critérios distintos a ter em consideração para a sua definição: a dimensionalidade, um problema pode ser unidimensional, bidimensional, tridimensional ou n -dimensional; o tipo de atribuição, onde o objetivo pode ser maximizar a produção (que ocorre quando o número de objetos disponíveis não são suficientes para alocar todos os itens), ou minimizar os recursos utilizados onde os objetos disponíveis são suficientes para alocar todos os itens, procurando-se posicionar os itens minimizando um determinado valor que pode ser, por exemplo, a quantidade de material desperdiçado; o sortido dos itens que podem ser idênticos, pouco heterogêneo ou muito heterogêneo; o sortido dos objetos, podendo existir um único objeto com todas as dimensões fixas ou com uma ou mais dimensões variáveis, ou muitos objetos idênticos, pouco heterogêneo ou muito heterogêneo; e a forma dos itens que pode ser regular ou irregular.

Para a classificação de PP segundo Wäscher et al. [2007] são necessárias 3 etapas: na primeira etapa combinam-se os critérios tipo de atribuição e tipo de itens, dando origem a **problemas básicos**; a segunda é definida pela combinação do tipo de objetos com os problemas básicos dando origem aos **problemas intermédios**; e, finalmente, na terceira etapa, são adicionados os critérios da dimensionalidade, a forma dos itens e a estrutura do problema, resultando em **problemas refinados**.

De seguida será feita uma breve referência a cada uma destas etapas, definindo alguns problemas que pertencem a cada um dos problemas básicos, intermédios e, finalmente, refinados.

Os **problemas do tipo básico** podem ser divididos em duas categorias consoante o tipo de atribuição ou objetivo usado na atribuição, que por sua vez se subdividem consoante a variedade e o sortido dos itens usados

(Figura 2.3).

- **Maximização da Produção**

Neste tipo de problemas os objetos existem em quantidades limitadas, não sendo possível a produção de todos os itens. Como se pretende maximizar a produção de itens, todos os objetos serão utilizados, resultando num problema de seleção dos itens a serem produzidos. Incluem-se nesta categoria:

- **Problemas de Posicionamento de Itens Idênticos**

Nesta categoria, pretende-se posicionar o maior número possível de itens, todos idênticos, num conjunto limitado de objetos. Deve notar-se que, como os itens são todos idênticos, não existe, na realidade, um problema de seleção de itens nem de agrupamento ou distribuição. Nesta categoria os problemas reduzem-se à determinação do melhor padrão em cada objeto, figura 2.4(a).

- **Problemas de Colocação**

Incluem-se nesta categoria os problemas onde o sortido de itens existente é fracamente heterogéneo e os objetos são limitados. O objetivo passa por maximizar a produção de itens ou por minimizar o desperdício (espaço nos objetos que não é ocupado por qualquer item). Nesta categoria, e contrariamente ao que acontece na anterior, já está implícito um problema de seleção dos itens, isto é, torna-se necessário decidir que itens posicionar, figura 2.4(b).

- **Problema da Mochila**

Tal como na categoria anterior existe uma seleção de itens que são caracterizados por um sortido muito heterogéneo a serem alocados num conjunto limitado de objetos. O objetivo é maximizar a produção de itens, figura 2.4(c).

- **Minimização dos Recursos Utilizados**

Estes problemas caracterizam-se pela existência de objetos suficientes para posicionar todos os itens. Nestes problemas, o objetivo usado na afetação é a minimização dos recursos utilizados, isto é, pretende-se minimizar o valor dos objetos necessários para alocar todos os itens.

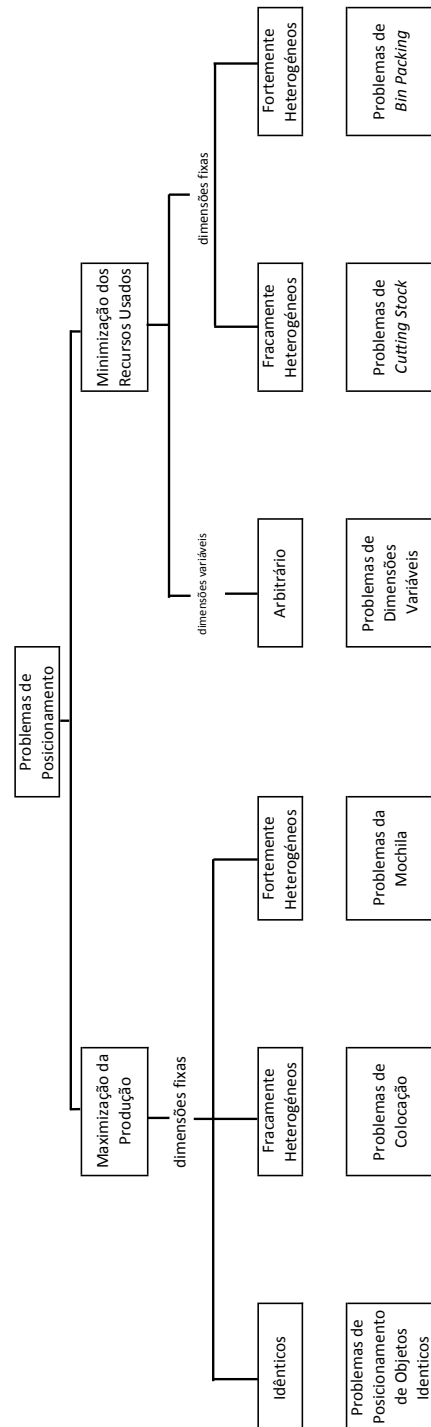


Figura 2.3: Categorias de Problemas de Posicionamento (adaptado de Wäscher et al. [2007].)

– **Problemas de Dimensões Variáveis**

A característica comum a estes problemas é a existência de uma ou mais dimensões variáveis do objeto ou objetos. O objetivo nestes problemas é posicionar todos os itens minimizando a dimensão ou dimensões variáveis do objeto, figura 2.4(d).

– **Problemas de *Cutting Stock***

Os problemas desta categoria incluem um sortido de itens pouco heterogéneo e um conjunto de objetos no qual é possível posicionar todos os itens. O objetivo é a minimização do número de objetos necessários para posicionar os itens na sua totalidade, ou minimizar o valor total dos objetos utilizados, figura 2.4(e).

– **Problemas de *Bin Packing***

A principal característica destes problemas é a existência de uma forte heterogeneidade do sortido dos itens, que devem ser posicionados num conjunto de objetos suficientes para o posicionamento completo dos itens. Nestes problemas pretende-se minimizar o número de objetos ou o valor desses objetos, figura 2.4(f).

Os **problemas de tipo intermédio** resultam dos problemas básicos por adição de um novo critério a ter em consideração: o **sortido dos objetos**. Nesta categoria continuam a existir os mesmos dois tipos de objetivos usados na afetação: a maximização da produção e minimização dos recursos utilizados.

No caso da **maximização da produção** os itens podem ser idênticos, fracamente heterogéneos e fortemente heterogéneos. No caso de itens idênticos pode ter-se um único objeto sendo estes problemas classificados como **Problemas de Posicionamento em Objetos Idênticos**. No caso de existirem itens fracamente heterogéneos e:

- um único objeto - **Problema de Colocação com um único objeto;**
- objetos idênticos - **Problema de Colocação com vários objetos idênticos;**
- objetos heterogéneos - **Problema de Colocação com vários objetos diferentes.**

No caso de haver itens fortemente heterogéneos e existir:

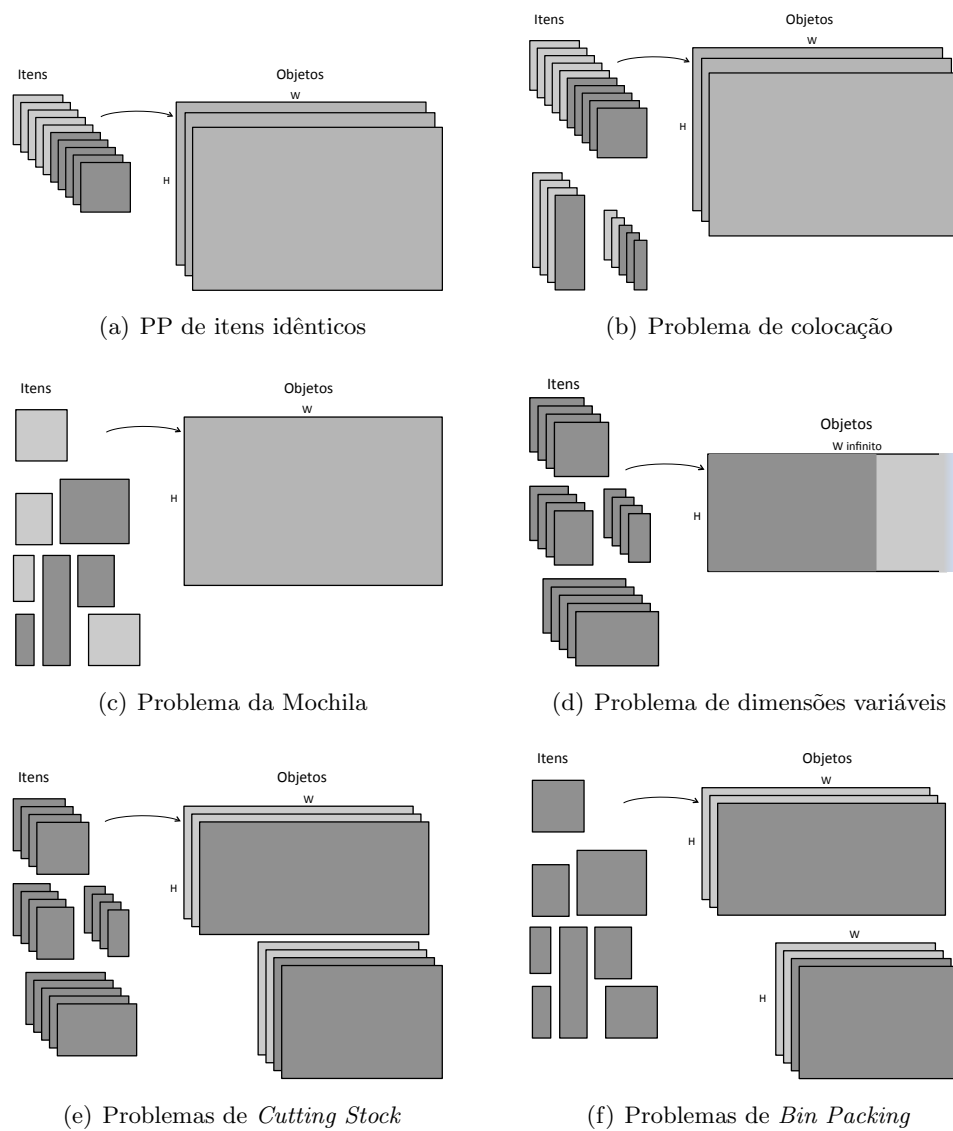


Figura 2.4: Problemas de posicionamento (adaptado de Oliveira et al. [2011a])

- um único objeto - **Problema da Mochila única**;
- objetos idênticos - **Problema de Mochilas idênticas**;
- objetos heterogêneos - **Problema de Mochilas diferentes**.

Relativamente à **minimização dos recursos utilizados**, os itens podem ser fracamente heterogêneos e fortemente heterogêneos. Se os itens existentes forem fracamente heterogêneos e houver:

- um único objeto - **Problema de *Cutting Stock* com objetos idênticos**;
- objetos idênticos - **Problema de *Cutting Stock* com múltiplos objetos**;
- objetos heterogêneos - **Problema de *Cutting Stock* Residual**;
- objetos com uma ou mais dimensões variáveis - **Problema de Dimensões Variáveis**.

No caso de haver itens fortemente heterogêneos e:

- um único objeto - **Problema de *Bin Packing* com objetos idênticos**;
- objetos idênticos - **Problema de *Bin Packing* com múltiplos objetos**;
- objetos heterogêneos - **Problema de *Bin Packing* Residual**;
- objetos com uma ou mais dimensões variáveis - **Problema de Dimensões Variáveis**.

Finalmente, os **problemas de tipo refinado** são obtidos pela aplicação dos critérios de dimensionalidade e da forma dos itens. A regra para classificação dos PP é obtida do seguinte modo: 1,2,3-D retangular, circular,..., irregular classificação intermédia. Por exemplo, a designação *2D retangular Open Dimensional Problem* refere-se a um PP no qual o objeto e os itens são retangulares e o objeto possui uma ou as duas dimensões variáveis.

Na literatura pode encontrar-se um vasto número de publicações relacionadas com PP que vão desde unidimensionais a tridimensionais, de itens retangulares a itens irregulares e que variam na abordagem de resolução.

Em todos estes problemas o principal objetivo é posicionar, sem sobreposições, um conjunto de itens de modo a maximizar a utilização do material e, consequentemente, a minimização do desperdício.

De seguida é feita uma breve descrição das categorias e subcategorias existentes nos PP de tipo refinado. Para além desta descrição, serão apresentados exemplos de problemas que podem ser encontrados em cada uma delas e referidos alguns autores e abordagens para estes problemas. Será dada preferência a problemas onde os itens não sejam retangulares, uma vez que a secção 2.2 é dedicada à apresentação de problemas e abordagens que lidam com itens com formas retangulares e ortogonais.

- **Maximização dos Produtos Produzidos**

- **Problema de Posicionamento de Itens Idênticos**

Nesta subcategoria existe um conjunto de itens idênticos a ser posicionado numa quantidade de objetos ilimitada. Um exemplo representativo desta categoria é o problema de carregamento de paletes, no qual uma paleta única deve ser carregada com um número máximo de caixas idênticas. Este problema foi abordado por Dowsland [1987] e por Morabito e Morales [1998] que apresentam algoritmos de programação dinâmica baseado num procedimento recursivo para a sua resolução. Para lidarem com o mesmo problema Alvarez-Valdes et al. [2005] usam um algoritmo baseado em Pesquisa Tabu.

Outro problema que pode ser incluído nesta subcategoria é o posicionamento de um conjunto de cilindros, com diâmetros idênticos, numa caixa retangular. Este problema foi abordado por Birgin et al. [2005a] através de um procedimento que se baseia no facto de os centros dos cilindros terem de estar dentro da caixa retangular definida pela base do contentor e distantes uns dos outros pelo menos um diâmetro. Com esta regra tentam encontrar o número máximo de centros de cilindros que satisfazem essas restrições. O mesmo problema foi abordado por Correia et al. [2000] que consideram o posicionamento dos cilindros em camadas. Estes autores conseguiram estender este problema à consideração de círculos com raios diferentes.

- **Problemas de Colocação num Único Objeto**

Esta subcategoria caracteriza-se pela existência de um único objeto disponível para posicionar um conjunto de itens fracamente heterogêneo. Podem aqui ser incluídos os problemas de posicionar um conjunto de círculos, de raios diferentes, num objeto retangular ou, por exemplo, problemas de carregamento de contentores onde se pretende posicionar um conjunto de caixas, de poucos tipos diferentes, num único contentor.

Outro problema surge quando se pretende cortar um conjunto de itens irregulares de um objeto também irregular. Crispin et al. [2003] apresentam uma abordagem baseada em algoritmos genéticos para este problema em que o objeto é de pele natural. Os itens são componentes para o fabrico de sapatos. O algoritmo apresentado consegue lidar com a não convexidade da forma dos itens, com a forma irregular do objeto e com restrições relativas à qualidade da pele. Sharma et al. [1997] tratam o mesmo problema considerando que o objeto para posicionamento é retangular.

– **Problemas de Colocação com Vários Objetos Idênticos**

Nesta subcategoria incluem-se problemas onde existem vários objetos disponíveis para posicionamento, todos com as mesmas dimensões. Podem aqui ser incluídos problemas similares aos da subcategoria anterior, considerando vários objetos idênticos. Por exemplo, considerando placas com a mesma dimensão para posicionamento, considerando paletes de iguais dimensões, etc.

– **Problemas de Colocação com Vários Objetos Diferentes**

Esta subcategoria caracteriza-se pela existência de vários objetos com diferentes dimensões. Os PP desta subcategoria podem ser obtidos pelos problemas das anteriores, considerando a existência de objetos diferentes disponíveis para posicionamento.

Um problema unidimensional que pertence a esta subcategoria pode ser encontrado em Gradišar et al. [1999] onde um sortido de comprimentos tem que ser cortado de um conjunto de comprimentos de entrada, todos diferentes. Neste problema, os comprimentos de entrada não são suficientes para satisfazer todas as encomendas, por conseguinte, o objetivo é a minimização do tamanho total dos comprimentos de pedidos brutos.

Um exemplo de problema tridimensional, que pode ser aqui in-

cluído, surge no carregamento de contentores quando há um conjunto de caixas (itens) que tem que ser posicionado num conjunto de contentores (objetos) de diferentes tamanhos. O espaço disponível nos contentores não é suficiente para acomodar todas as caixas. Assim, tem que ser determinada uma seleção de caixas que maximiza a utilização do volume do contentor ou, alternativamente, o valor das caixas posicionadas.

– **Problema da Mochila Única**

No problema clássico unidimensional da Mochila única pretende-se posicionar um conjunto de itens, com determinado peso e valor dentro de uma mochila, com uma determinada capacidade (limite de peso), de modo a maximizar o valor dos itens posicionados. Quando o peso dos itens for idêntico ao seu valor o Problema da Mochila designa-se por Problema da Soma dos Subconjuntos (Problema *Subset-Sum*). Extensões do problema Clássico unidimensional da Mochila resultam em considerar duas ou mais dimensões resultando nos problema da Mochila única bidimensional e tridimensional.

O Problema da Mochila Tridimensional onde um conjunto de caixas retangulares deve ser posicionado num contentor foi abordado por Bischoff e Marriott [1990] que distinguem casos onde uma carga deve ser transportada em diferentes contentores e casos onde o volume máximo de uma carga deve ser carregado num único contentor. Estes autores referem também casos onde uma carga deve ser colocada pelo seu peso e destino e casos onde o objetivo é obter uma utilização máxima do contentor como função do volume e valor da carga posicionada.

Bortfeldt e Gehring [2001] apresentam um algoritmo genético para a resolução do problema de carregamento de um contentor com caixas de diferentes tamanhos num único contentor. São geradas várias camadas verticais para o posicionamento das caixas.

Em ambos os casos bidimensional e tridimensional o valor do posicionamento dos itens deve ser maximizado. Se o valor dos itens for proporcional ao seu tamanho/volume pode dizer-se que é equivalente a minimizar o espaço vazio do objeto ou contentor.

O problema Clássico da Mochila foi estendido a n -dimensional por Fekete et al. [2007].

Esta categoria, tal como nas demais categorias, pode também incluir problemas em que as formas dos itens e dos objetos não são retangulares ou paralelepípedicos. Casos destes podem ser, por exemplo, encontrados em:

- * George et al. [1995] que apresentam o problema de posicionar um conjunto de itens circulares, de diferentes diâmetros, num objeto retangular cuja aplicabilidade surge no transporte de tubos. Foram tidas em consideração várias necessidades práticas tais como a estabilidade de *stock*, de operacionalidade e o máximo de utilização de espaço possível. Neste trabalho o problema é formulado como programação não-linear inteira mista e são usados métodos heurísticos para a resolução aproximada deste problema;
- * Lins et al. [2003] apresentam uma abordagem exata denominada em L com base numa função definida recursivamente para o posicionamento de retângulos em objetos retangulares e em forma de L com aplicação prática no carregamento de paletes. No entanto, esta abordagem requer um elevado tempo computacional. O problema de tempo computacional desta abordagem foi atenuado no trabalho realizado por Birgin et al. [2005b];
- * Esta categoria pode também ser encontrada em problemas cujos itens têm formas irregulares. Por exemplo, Heistermann e Lengauer [1995] consideram o problema de posicionar um conjunto de itens com formas irregulares num objeto, também irregular, de couro natural. O objetivo deste problema é a minimização do desperdício. Sendo o objeto um produto natural que apresenta formas e zonas com diferentes qualidades, que variam de objeto para objeto. Este problema foi também abordado por Tay et al. [2002], no entanto, o objeto não é de origem natural e, conseqüentemente, não apresenta quaisquer defeitos não sendo necessário considerar a divisão do objeto em diferentes regiões que se distinguem pela qualidade.

- **Problema de Mochilas Idênticas**

Estes problemas obtêm-se pela extensão dos problemas de Mochila anteriormente apresentados, considerando-se disponíveis para posicionamento várias Mochilas Idênticas. O objetivo é maximizar o número de itens posicionados.

Um exemplo desta subcategoria surge no trabalho de Raidl e Kodydek [1998] onde um conjunto de contentores idênticos deve ser preenchido por um conjunto de itens (caixas) com pesos e valores dados. O objetivo é maximizar o valor total dos itens posicionados. Como abordagem a este problema usam algoritmos genéticos.

- **Problema de Mochilas Heterogêneas**

Esta subcategoria é uma extensão da anterior considerando que as mochilas disponíveis têm características diferentes.

Martello e Toth [1990] e Pisinger [1999] consideram o problema Clássico da Mochila, que é um exemplo representativo desta categoria. Neste problema, um conjunto de itens fortemente heterogêneos, cada item caracterizado por um peso e um ganho, deve ser posicionado num conjunto de Mochilas com diferentes capacidades que não devem ser excedidas. O objetivo é maximizar o ganho dos itens posicionados.

- **Problemas de Posicionamento de Minimização de Recursos Utilizados**

- **Problema de *Cutting Stock* de Tamanho Único**

Nesta subcategoria os objetos são idênticos. Podem ser aqui incluídos problemas em que um conjunto de retângulos, fracamente heterogêneo, tem que ser cortado de um conjunto de placas todas iguais ou, onde um conjunto de caixas, fracamente heterogêneo, tem que ser posicionado no menor número de contentores iguais.

Pertence a esta subcategoria o problema considerado por Cui [2005], onde se pretende posicionar um conjunto de itens circulares, com o mesmo diâmetro, num conjunto de objetos retangulares. Como solução para este problema são gerados padrões em T com o objetivo de aumentar o aproveitamento dos objetos.

– **Problema de *Cutting Stock* com Vários Tamanhos**

Nesta subcategoria, os objetos são fracamente heterogêneos. Os problemas desta subcategoria são extensões dos problemas da subcategoria anterior, considerando, por exemplo, placas ou contentores de tamanhos diferentes.

Morabito e Garcia [1998] tratam um problema prático que se pode incluir nesta subcategoria. Este problema consiste no corte de chapas de fibra de madeira comprimidas (MDF). O objetivo consiste em determinar a melhor forma de cortar as chapas duras (objetos) de forma a produzir as unidades menores (itens) com a menor perda possível de material.

– **Problema de *Cutting Stock* Residual**

Esta subcategoria caracteriza-se pela existência de um conjunto de objetos fortemente heterogêneo. A principal característica dos problemas que pertencem a esta subcategoria, que os distingue dos demais problemas, é que as partes não usadas dos objetos em operações de corte anteriores resultam em novos objetos que podem ser usados em futuras operações de corte. Os problemas desta subcategoria resultam das duas subcategorias anteriores com sucessivas operações de corte para o reaproveitamento de partes de objetos não utilizadas.

Vanderbeck [2001] trata um problema, que pode ser incluído nesta subcategoria, que consiste no corte de itens retangulares em itens de largura e comprimento especificado. O processo de corte envolve três fases de corte ortogonal guilhotinável: as peças de *stock* são cortados em seções, que são cortadas em fendas que são cortadas em itens pedidos. O objetivo deste problema é a minimização do desperdício. Neste trabalho são tidas em consideração questões como, por exemplo, o envelhecimento dos itens em *stock*, a urgência dos pedidos e os custos de instalação. Vanderbeck [2001] apresenta uma abordagem que envolve a decomposição do problema e o uso recursivo o método de geração de colunas. São determinados limites inferiores com base no custo mínimo e, por arredondamento da solução obtida por programação linear, é obtida uma solução do problema admissível e um limite superior associado.

– **Problema de *Bin Packing* de Tamanho Único**

Nesta subcategoria os objetos existentes para posicionamento são todos iguais e estão disponíveis em quantidade ilimitada.

O Problema de *Bin Packing* de Tamanho Único pode ser encontrado no trabalho de van Stee [2004] onde é necessário posicionar um conjunto de itens com forma geométrica quadrada em objetos que são igualmente quadrados. O objetivo é minimizar o número de objetos usados. Como abordagem a este problema usam um algoritmo que vai posicionando os itens de acordo com o seu tamanho.

– **Problema de *Bin Packing* com Vários Tamanhos**

O que caracteriza esta subcategoria é a existência de objetos fracamente heterogêneos, disponíveis em quantidade ilimitada, com determinada capacidade e valor específico.

Okano [2002] descreve um algoritmo heurístico para o posicionamento bidimensional de um conjunto de itens de forma geométrica livre em objetos que também podem ter uma forma geométrica qualquer. O objetivo é minimizar o número de objetos usados e, para cada objeto, minimizar o desperdício. Os itens e os objetos são aproximados por segmentos de reta, sendo depois tratados como conjuntos de intervalos. O algoritmo de posicionamento usado é uma variante do algoritmo de ajuste decrescente de itens, estendido a partir do caso unidimensional para o bidimensional. É também usado um algoritmo de posicionamento de grupos de itens que é uma extensão do algoritmo de posicionamento com a recombinação de itens de entrada.

– **Problema de *Bin Packing* Residual**

Nesta subcategoria o sortido de objetos é fortemente heterogêneo. Um exemplo de PP que se enquadra nesta subcategoria é quando se pretende posicionar um conjunto de itens onde existe uma grande variedade objetos retangulares disponíveis para posicionamento.

Chen et al. [1995] consideram a versão tridimensional do problema de carregamento de contentores em que um conjunto de itens fracamente heterogêneo tem que ser posicionado em contentores, disponíveis em diferentes tamanhos. O espaço disponível

não é suficiente para o posicionamento de todas as caixas. Neste problema, existe um problema de seleção de caixas que tem que ser determinado para a maximização do volume ou, alternativamente, para a maximização do valor das caixas posicionadas.

Para além dos problemas anteriormente apresentados, onde um conjunto de itens tem que ser posicionado num conjunto de objetos com o objetivo de minimizar o desperdício, designados por PP puros, são também referidos na tipologia extensões e variantes a estes PP. São considerados variantes os problemas onde: a função objetivo é diferente da minimização do desperdício, onde é determinística; é *fuzzy* (contém coeficientes que são *fuzzy*); contém mais do que um objetivo; é inexistente, sendo apenas um problema de admissibilidade (de decisão). Em relação aos itens existentes, se são de tipos não homogêneos; se são não homogêneos em relação às quantidades; se a sua procura é feita relativamente a diferentes graus de qualidade; se não forem conhecidos à priori (problema *online*) também são considerados variantes. São igualmente variantes problemas onde existe, nos objetos, uma variedade de formas, isto é se houver objetos que não têm a mesma forma geométrica, ou se os objetos possuírem defeitos ou diferentes graus de qualidade.

Para além das variantes a PP há também problemas que são considerados extensões. Podem ser incluídos, por exemplo, no conjunto das extensões os problemas: com múltiplos períodos; que considerem que há diferentes tamanhos de lotes; onde há máquinas paralelas; em que há sequenciamento de padrões; onde existem peças residuais; de rotas de veículos; ou, por exemplo, onde se pretenda minimizar o número de padrões de corte.

2.1.2 Geometria em Problemas de Posicionamento

Tal como foi visto anteriormente, quando se pretende posicionar um conjunto de itens, num conjunto de objetos, tanto os itens como os objetos podem ter diversas formas geométricas. As propriedades geométricas dos problemas têm uma enorme influência na complexidade dos PP. Para além das dimensões espaciais e a forma dos objetos, a geometria dos itens, a variedade e o sortido dos itens são critérios muito importantes. Paralelamente às características do problema há diferentes restrições que podem ser impostas aos padrões e aos itens.

Nesta subsecção será feita uma referência às diferentes formas que os

itens podem ter e o que é que os distingue e, serão descritas algumas restrições que podem impostas aos padrões resultantes do posicionamento dos itens. Uma vez que a dimensionalidade do problema abordado neste trabalho é bidimensional será feita apenas referência à geometria de PP bidimensionais.

No que diz respeito às dimensões, os itens podem ter dimensões fixas (*hard itens*) ou área fixa e dimensões que variam dentro de um determinado intervalo (*soft itens*). Os itens podem também diferir de acordo com a sua forma, sendo uma das principais características dos PP pois tem uma enorme influência na complexidade das considerações geométricas destes problemas. Geometricamente, os itens podem ser divididos em regulares e não regulares, podendo ainda, para cada caso, serem convexos ou não convexos. Os itens regulares são aqueles cujos lados e ângulos internos são congruentes, isto é, são geometricamente iguais, figura 2.5(a), caso contrário dizem-se não regulares, figura 2.5(b). Os polígonos convexos são constituídos por segmentos de reta cujo ângulos internos entre esses segmentos não ultrapassam 180° . Considerando qualquer segmento de reta com extremidades no interior destes polígonos todos os pontos do segmento pertencem ao interior do polígonos, ambos os polígonos representados nas figuras 2.5(a) e 2.5(b) são convexos. No caso dos polígonos não convexos, os ângulos internos entre os segmentos de reta que o formam podem ser superiores a 180° e, se for traçado um segmento de reta com extremidades no seu interior pode ter pontos que não pertençam ao interior do polígono, figura 2.5(c).

Para assegurar a não sobreposição dos itens a variedade e o sortido dos itens tem uma enorme importância na complexidade dos PP. Um PP fica bastante mais simples quando existe apenas um tipo de item, mesmo que tenham diferentes dimensões, do que quando existe uma variedade de itens, isto é, itens com diferentes formas. Por exemplo, para lidar com a componente geométrica em problemas onde todos os itens são retângulos, é quase imediato, uma vez que basta verificar se os itens estão separados na horizontal e se estão separados na vertical, sendo esta verificação independente uma da outra. Outro caso simples resulta do posicionamento de itens todos circulares onde, para assegurar que não há sobreposições, basta verificar se os centros distam a soma dos raios. Caso exista uma grande variedade de itens a complexidade dos problemas cresce de forma polinomial com o aumento do número de itens a posicionar e com o aumento do número de vértices de



Figura 2.5: Polígonos regulares e não regulares, convexos e não convexos

cada item.

A forma como os itens estão posicionados no objeto é designada por padrão (ou padrão de corte). A solução de um PP é representada por um ou mais padrões que resultam do posicionamento de um conjunto de itens, sem sobreposições, de modo a otimizar um determinado objetivo. Para além das considerações geométricas anteriormente apresentadas, um outro aspeto a ter em consideração é a geometria dos padrões. Os padrões resultantes podem ser guilhotináveis ou não guilhotináveis. Um padrão diz-se guilhotinável quando puder ser obtido por sucessivos cortes horizontais ou verticais de um lado ao outro do padrão que o divide em dois retângulos, figura 2.6(a), caso contrário diz-se não guilhotinável, figura 2.6(b).

Relativamente ao posicionamento de polígonos ortogonais, se são posicionados com os lados paralelos aos eixos vertical e horizontal diz-se que o padrão gerado é ortogonal, ambos os padrões representados na figura 2.6 são ortogonais, contrariamente ao que acontece com o padrão da figura 2.7 que é não ortogonal.

Se o número de itens a posicionar for elevado, mas o sortido de itens for reduzido pode justificar-se um posicionamento periódico, onde os itens são replicados de uma forma regular ou periódica, figura 2.8.

Outra restrição geométrica que pode ser imposta aos itens está relacionada com a sua rotação. Os itens podem ter orientação fixa, isto é, não podem sofrer qualquer rotação ou, caso contrário, podem ser rodados segundo determinado ângulo dizendo-se que têm rotação permitida e x graus. Na figura 2.9(a) está representado um item disponível para posicionamento. Se tiver orientação fixa tem que ser posicionado na sua orientação original, 2.9(b), caso possa rodar, e o posicionamento seja ortogonal, pode ser posicionado com a orientação da figura 2.9(b) ou da figura 2.9(c). No caso de

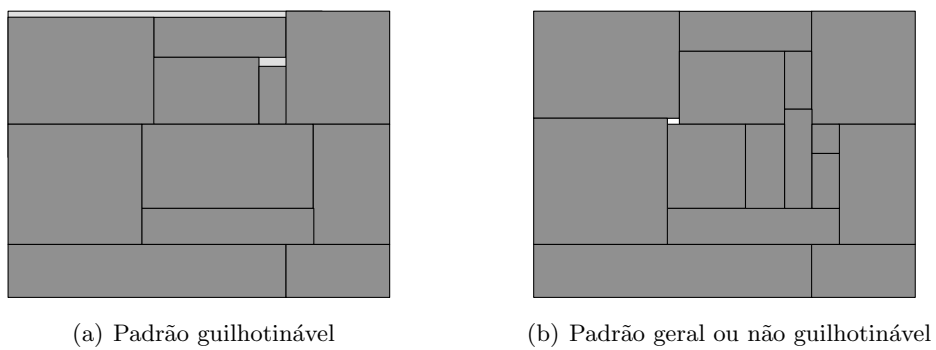


Figura 2.6: Padrão guilhotinável e não guilhotinável

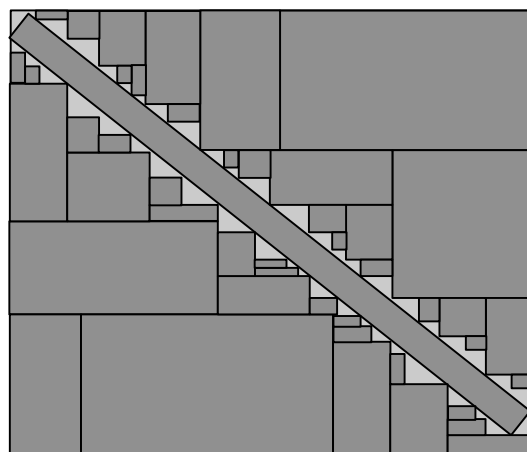


Figura 2.7: Exemplo de um padrão não ortogonal

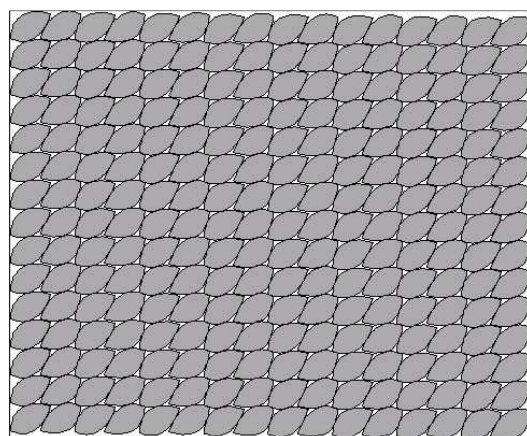


Figura 2.8: Exemplo de um padrão periódico (adaptado de Costa et al. [2009])

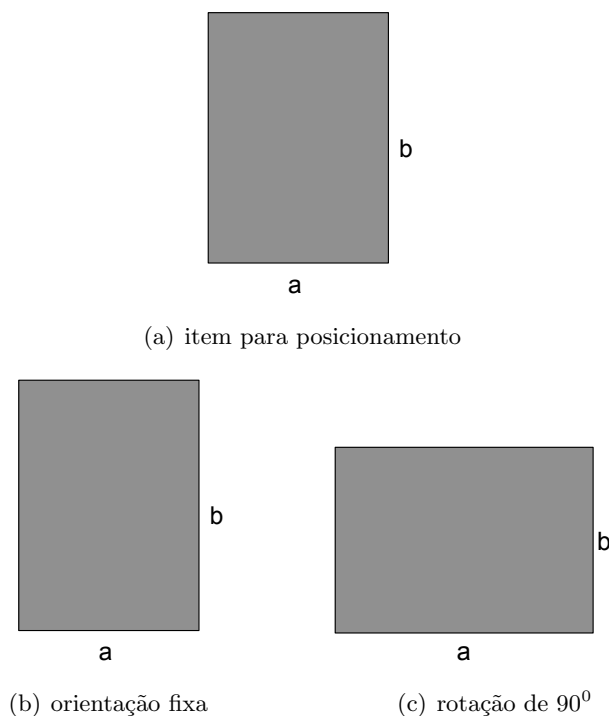


Figura 2.9: Itens com orientação fixa e orientação de x graus

existirem para posicionar itens retangulares e o posicionamento permitido seja ortogonal, os itens só podem rodar 0 ou 90^0 , se os itens forem ortogonais podem sofrer rotações de 0 , 90^0 , 180^0 ou 270^0 . A rotação ou não dos itens apenas depende das características do problema. Por exemplo, na indústria têxtil, não são permitidas rotações de itens quando os objetos têm texturas ou desenhos que diferem nas diferentes direções.

Neste trabalho os padrões resultantes do posicionamento de itens são ortogonais e os itens são polígonos ortogonais. Definem-se polígonos ortogonais aqueles cujos ângulos internos são 90^0 ou 270^0 , por exemplo itens com formas em L, T, U, itens 8, 4 e 5 da figura 2.2, respectivamente. As dimensões dos itens são conhecidas e os itens podem sofrer rotações de 0 , 90^0 , 180^0 ou 270^0 . No caso particular dos itens serem retangulares as rotações permitidas são 0 ou 90^0 .

2.2 Problemas de Posicionamento de Retângulos em Objetos Definidos

As publicações relacionadas com PP que foram surgindo apresentam, para além de variações no tipo de objetivo da afetação, nas dimensões dos objetos e no sortido dos itens, resultando em diferentes categorias de PP, variações na formulação e nas abordagens usadas para solucionar o problema.

Nesta secção será feita uma referência a PP bidimensionais em que os itens são retangulares e o objeto ou objetos, disponíveis para posicionamento, são conhecidos à partida, isto é, as suas dimensões são conhecidas. Será feita referência a Problemas de Colocação, Problemas de *Cutting Stock*, Problema da Mochila, Problemas de *Bin Packing* e problemas de *Strip Packing*. Começa-se por uma introdução histórica acerca do aparecimento destes problemas, seguindo-se uma pequena seleção de trabalhos e abordagens existentes para PP. Uma revisão mais detalhada acerca destes problemas pode ser encontrada em Wäscher et al. [2007].

O primeiro trabalho na área de PP foi realizado por Kantorovich [1960] cuja publicação foi feita em russo em 1939, sendo publicada em inglês apenas em 1960. Neste trabalho são apresentados modelos matemáticos de programação linear para o planeamento e organização da produção.

Um marco importante foi o trabalho de Gilmore e Gomory [1961] que usaram técnicas de programação linear para a obtenção de soluções ótimas aplicadas problemas unidimensionais de *Cutting Stock*. No entanto, apenas conseguem resolver instâncias de pequena dimensão dado o elevado tempo computacional necessário. Para colmatar o elevado número de padrões que é necessário gerar usaram mais tarde [Gilmore e Gomory, 1963] técnicas de geração de colunas para a solução da relaxação linear do problema. O trabalho desenvolvido por estes autores foi aplicado a problemas reais que consistiam no corte de rolos na indústria de papel. Em 1965, conseguiram estender o seu trabalho a problemas bidimensionais [Gilmore e Gomory, 1965]. No entanto, dado o elevado número de colunas que geravam padrões não admissíveis, impuseram a restrição de obter apenas padrões guilhotináveis. Estes autores usaram também técnicas de programação dinâmica para a resolução de Problemas da Mochila [Gilmore e Gomory, 1966].

Desde o trabalho de Gilmore e Gomory [1966] vários autores têm publicado artigos para PP bidimensionais em que os itens são retângulos.

No problema de *Cutting Stock* está disponível para posicionamento um conjunto de itens fracamente heterogêneo e o objetivo é a minimização de recursos utilizados. Cemil Dikili [2004] aborda este problema considerando para posicionamento um conjunto de objetos idênticos e para solução padrões guilhotináveis. Apresenta uma abordagem onde um algoritmo considera seletivamente padrões admissíveis através da eliminação de arranjos de itens que tornam o problema irresolúvel. O mesmo problema foi tratado por Faina [1999] que propôs dois algoritmos, um que envolve restrições de guilhotina e outro para padrões gerais. Ambos os algoritmos se baseiam no algoritmo de arrefecimento simulado.

Uma outra subcategoria de PP é o Problema da Mochila, onde um conjunto de itens fortemente heterogêneo deve ser posicionado, num ou vários objetos, com o objetivo de maximização dos produtos produzidos.

O Problema da Mochila única foi abordado por Fekete et al. [2007] que desenvolveram um algoritmo para provar que um conjunto de itens pode ser posicionado dentro de um dado objeto utilizando características e propriedades de grafos de intervalo. Mais tarde, Ferreira e Oliveira [2008] introduziram uma nova propriedade dos grafos de intervalo que permitiu gerar um padrão geral. Hadjiconstantinou e Christofides [1995] apresentaram uma abordagem exata baseada numa pesquisa em árvore para o mesmo problema (Mochila Única). Limitaram o número de vezes que um item pode ser usado num padrão. Foram usados limites, obtidos a partir da relaxação Lagrangiana, para limitar a expansão da pesquisa. Para a otimização deste limite usam subgradientes. Wei et al. [2009] consideram o problema de posicionamento bidimensional com o objetivo de maximizar a área preenchida pelos retângulos posicionados (Problema da Mochila). Para o posicionamento dos itens consideram a mesma estratégia apresentada em Martello et al. [2003]. Esta estratégia baseia-se no conceito de invólucro, no entanto, apresentam uma melhoria pois consideram menos pontos admissíveis do que os usados em Martello et al. [2003]. Para pontos possíveis de posicionamento de itens apenas são considerados os pontos do invólucro que mudam de segmentos de reta verticais para horizontais. Para verificarem se um item pode ser posicionado num ponto admissível apenas têm que verificar se os limites do objeto são ultrapassados.

Os Problemas de Colocação são outra subcategoria dos PP que se caracterizam pela existência de um conjunto de itens, fracamente heterogêneo,

onde o objetivo é a maximização dos produtos produzidos. Vários autores abordaram este problema considerando que os itens são retângulos, entre os quais Herz [1972] e Beasley [1985] que desenvolveram métodos exatos de resolução para gerar padrões não guilhotináveis restritos em problemas bidimensionais. O método usado baseia-se numa pesquisa em árvore controlada, que usa limites obtidos através de relaxação Lagrangiana e que são otimizados através de uma otimização por subgradiente. Para estes autores o número de vezes que os itens podem ser usados no padrão de corte não é limitado. A versão restrita deste problema foi também resolvida por Christofides e Whitlock [1977] e Christofides e Hadjiconstantinou [1995]. Christofides e Whitlock [1977] usaram uma pesquisa em árvore para a resolução, de forma exata, de problemas de *Cutting Stock* guilhotináveis, com a limitação do número de vezes que um item pode ser usado no padrão. Em 1995, os mesmo autores [Christofides e Hadjiconstantinou, 1995] desenvolveram uma abordagem baseada numa pesquisa em árvore que resolve problemas guilhotináveis de dimensão média. Este problema foi também abordado por G et al. [2003] que tratam o problema de encontrar o melhor padrão guilhotinável que resulta do posicionamento de itens retangulares, com orientação fixa, num objeto retangular, de forma a maximizar a soma total dos lucros. Cada item tem um tamanho e um lucro associado. No caso de problemas não ponderados, onde os lucros das peças não são dados, usam a área dos itens. Este autores propõem um algoritmo *branch-and-bound* baseado na técnica do primeiro melhor ramo. Esta abordagem, ao contrário da maioria dos algoritmos exatos existentes, trata o problema de baixo para cima, usando construções horizontais e verticais. As estratégias de ramificação usadas permitem uma redução do número de nós que são pesquisados através da eliminação de padrões dominados. Wang [1983] desenvolveu dois algoritmos exatos para o problema de corte restrito. Estes métodos juntam itens que formam boas combinações para gerar cortes guilhotináveis. A solução ótima é obtida gerando todas as combinações possíveis. Este autor usa um limite máximo de percentagem de desperdício e limites que medem o erro através da distância entre o desperdício do padrão gerado e o da solução ótima. Mais tarde, Oliveira e Ferreira [1990] melhoram o algoritmo proposto por Wang [1983] usando o método proposto por Gilmore e Gomory [1966] para a modificação do critério de aceitação para as combinações possíveis de itens redefinindo o limite de percentagem de desperdício.

Outra subcategoria dos PP são os Problemas de *Cutting Stock* onde existe um conjunto de itens fracamente heterogêneo e um conjunto de objetos disponíveis para o posicionamento. Mornar e Khoshnevis [1997] tratam o problema de *Cutting Stock* com múltiplos objetos fracamente heterogêneos. Neste trabalho é descrito um sistema integrado, que inclui uma base de dados para os itens e objetos iniciais, inclui um procedimento de corte de *stock*, baseado num algoritmo heurístico, e uma produção do planeamento que usa programação linear.

Os Problemas de *Bin Packing*, que são caracterizados pela existência de um conjunto de itens fortemente heterogêneo e onde o objetivo é a minimização dos recursos utilizados, são outra subcategoria dos PP. Pode ser encontrada em [Lodi et al., 2002] uma revisão bibliográfica sobre estes problemas, dando ênfase a algoritmos exatos e abordagens baseadas em heurísticas e meta-heurísticas. Neste trabalho são também discutidos limites inferiores e superiores e algumas variantes, resultantes de aplicações práticas, de PP relacionadas com problemas de *Bin Packing*. Este problema foi tratado por Martello e Vigo [1998] considerando disponível para posicionamento um conjunto de objetos idênticos. Estes autores, propõem uma abordagem exata para este problema baseada no algoritmo *branch-and-bound*, considerando novos limites inferiores. Em Hopper e Turton [2001] pode ser encontrado um estudo que investiga a performance de técnicas evolucionárias e outros métodos de pesquisa baseados em meta-heurísticas para problemas de *Bin Packing*.

Um outro problema que surge no posicionamento de itens é o Problema de *Strip Packing*. Este problema é muito semelhante ao problema de *Bin Packing* diferindo apenas no objeto disponível para o posicionamento dos itens. Enquanto nos problemas de *Bin Packing* os objetos possuem dimensões fixas, nos problemas de *Strip Packing* o objeto tem uma das dimensões infinita, sendo o objetivo a minimização dessa dimensão. Estes problemas são problemas de dimensões variáveis de acordo com a tipologia de PP. Hifi [1998] propôs um algoritmo exato para gerar padrões guilhotináveis em problemas de *Strip Packing* baseado na técnica *branch-and-bound*. Este algoritmo é capaz de resolver apenas instâncias de pequena dimensão. Hopper e Turton [2001] resolvem problemas retangulares bidimensionais onde um conjunto de itens tem que ser posicionado num único objeto. Estes autores propõem uma combinação de meta-heurísticas com o algoritmo de posicio-

namento *bottom-left*. Neste artigo é também apresentada uma revisão bibliográfica de diferentes abordagens existentes com recurso a meta-heurísticas para problemas de *Strip Packing*. Iori et al. [2003] propõem, para o problema de *Strip Packing*, um algoritmo de pesquisa Tabu, um algoritmo genético e uma combinação destes dois algoritmos numa abordagem híbrida. Martello et al. [2003] propuseram, como abordagem ao problema de *Strip Packing*, um algoritmo *branch-and-bound* onde são descritos e usados limites inferiores que se baseiam em considerações geométricas e limites obtidos através de relaxação do problema. Estes autores conseguem resolver instâncias até 200 itens. Lesh et al. [2004] apresentaram uma abordagem exaustiva para o problema de *Strip Packing* onde os posicionamentos não contêm desperdício de espaço. Este problemas são designados por problemas de posicionamento perfeitos ou *jigsaw puzzles*. Como abordagem usaram técnicas *branch-and-bound* melhoradas com programação dinâmica. Usam a regra de posicionamento *bottom-left* dentro do algoritmo *branch-and-bound*. Pisinger e Sigurd [2005] tratam o problema de *Strip Packing* bidimensional de forma exata. Neste problema, os objetos têm custos variáveis (custos que não são proporcionais ao tamanho do objeto) e o objetivo é minimizar o custo dos objetos usados. São apresentados limites inferiores para o problema, um dos quais baseado na decomposição de Dantzig-Wolfe usado no algoritmo *branch-and-price*. Bortfeldt [2006] propõe um algoritmo genético para o problema de *Strip Packing* guilhotinável. Neste trabalho, a pesquisa opera diretamente no padrão, isto é, sem o recurso a soluções codificadas. Os padrões são manipulados através de operadores genéticos específicos. Alvarez-Valdes et al. [2008] propuseram um algoritmo *branch-and-bound* obtido através da hibridização entre o algoritmo GRASP e um algoritmo exato para a resolução do problema bidimensional *Strip Packing*.

Vários autores usam abordagens que combinam algoritmos que envolvem a geração de sequências, numa primeira fase, que são de seguida decodificadas, posicionando os itens um a um de acordo com a sequência, por algoritmos de posicionamento. Os algoritmos de posicionamento mais usados são:

- *bottom-left* onde os itens têm que ser posicionados na posição mais à esquerda e mais abaixo (depois de todos os itens posicionados não é possível mover qualquer item para esquerda ou para baixo sem sobreposições), este algoritmo foi proposto por Baker et al. [1980] e mais

tarde melhorado por Liu e Teng [1999];

- caso o item seja posicionado no ponto mais à esquerda do conjunto de pontos de posicionamento mais abaixo designam-se por *bottom-left-fill* proposto por Chazelle [1983];

2.3 Problemas de Posicionamento de itens de Minimização da Área do Objeto

O problema base abordado nesta dissertação refere-se ao posicionamento de um conjunto de itens, sem sobreposições, num objeto que não é conhecido à partida. O objetivo deste problema é a minimização da área do retângulo exterior envolvente que contém todos os itens, isto é, pretende-se minimizar a área do objeto (este objetivo será daqui em diante designado apenas por minimização da área). Este problema difere dos problemas anteriormente apresentados uma vez que o objeto não tem nenhuma dimensão definida.

Nesta secção será feita referência a publicações existentes na área de PP bidimensionais que envolvem itens retangulares e polígonos ortogonais onde o objetivo seja a minimização da área.

2.3.1 Problemas com Itens Retangulares

O PP de minimização de área surge no posicionamento de módulos em circuitos integrados, Very Large Scale Integration (VLSI), para os quais encontrar o melhor posicionamento dos módulos para obter circuitos compactos é um objetivo importante. Um dos principais aspetos desta miniaturização é o tamanho total que determina o custo de fabrico. Distribuir um maior número de componentes numa área menor é a melhor maneira de reduzir os custos do circuito e aumentar a velocidade. Uma característica das instâncias que surgem nesta aplicação real é o elevado número de itens para posicionar, podendo chegar aos milhares. A forma dos itens pode ser retangular ou ortogonal e as suas dimensões podem ser fixas ou ter área fixa e dimensões que podem variar dentro de um determinado intervalo. Esta última característica é muito frequente nesta aplicação porque os itens, também denominados por módulos, contêm no seu interior um conjunto de componentes que podem ser dispostos de várias formas mas ocupam uma área fixa. O foco deste trabalho é lidar com itens ortogonais com dimensões conhecidas, não sendo

tratados problemas onde os itens tenham dimensões variáveis.

Pode mesmo dizer-se que o problema de posicionamento de polígonos ortogonais tem como uma das principais aplicações o design de circuitos *VLSI*, sendo que a maioria das publicações existentes, relacionadas com este problema, referem-se a este domínio de aplicação.

Nesta aplicação, o elevado número de itens para posicionamento implica, muitas vezes, o uso de heurísticas ou meta-heurísticas, para explorar o espaço de soluções. Para facilitar pesquisas e representar padrões, existem duas formas distintas de representar as posições entre itens, uma em que as posições dos itens são especificadas por coordenadas definidas em relação a um sistema de eixos, e outra que é definida por um conjunto de codificações, onde cada configuração representa um padrão. Uma codificação representa, usualmente, as relações topológicas entre itens, isto é, as posições relativas entre pares de itens. As representações podem ser baseadas em grafos ou baseadas nos padrões. As representações baseadas em grafos incluem grafos de restrições, árvores binárias, grafos de vizinhança, *O-tree*, *B*-tree*, etc. Os grafos de restrições são representações básicas que indicam as posições relativas entre itens; os grafos de vizinhança, são uma simplificação dos grafos de restrições, têm apenas em consideração as quatro vizinhanças de cada item retangular; a árvore binária reduz esta informação apenas a duas vizinhanças para cada item; tanto a *O-tree* como a *B*-tree* são uma simplificação da árvore binária contendo apenas uma árvore para a codificação, diferindo principalmente entre elas no número de ramos que a árvore contém. Podem também ser incluídas no conjunto das representações que indicam a posição relativa entre pares de itens as representações: *sequence pair*, a *bounded slicing grid*, a árvore guilhotinada e *transitive closure graph*. A primeira representação foi proposta por Wong e Liu [1986] que apresenta uma estrutura em árvore para representar padrões guilhotináveis. Nesta árvore, as folhas representam um item e cada nó interno representa o tipo de corte utilizado, horizontal ou vertical, para unir dois itens (figura 2.10). Dada uma árvore que represente um padrão guilhotinável, a sua *Normalized Polish Expression* é uma sequência de n operandos (itens) e $n - 1$ operadores (H, V). Esta sequência de nós visitados pode ler-se começando de baixo para cima, na árvore. No entanto, dado um padrão guilhotinável a correspondente árvore (e, consequentemente, a *polish expression*) não é única, existindo redundância neste tipo de representações. Numa *Normalized Polish Expression* não

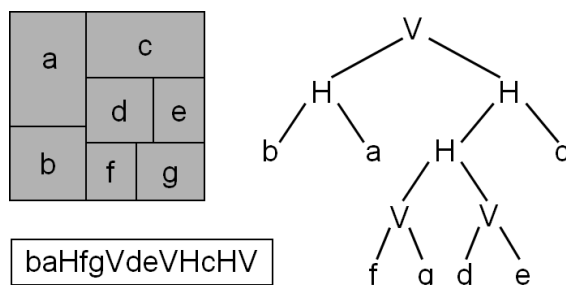


Figura 2.10: Árvore que representa um padrão guilhotinável e a correspondente *Normalized Polish Expression*.

existem operadores iguais consecutivos. Quando a representação for usada numa abordagem de melhoramento, as perturbações usadas são troca de 2 itens, troca do operador V por H e H por V em toda a sequência e a troca dois operandos e dois operadores.

De todas as representações que foram surgindo ao longo dos tempos, um marco importante foi o aparecimento da primeira representação para padrões gerais (não guilhotináveis) proposta por Murata et al. [1995]. Esta representação baseia-se num par de sequências, *sequence pair* (SP), para especificar posições relativas entre retângulos. Muitas outras representações surgiram após o SP: *bounded slicing grid* (Nakatake e Fujiyoshi [1997]), *O-tree* (Yoshimura [1999]), *B*-tree* (Chang et al. [2000]), *transitive closure graph* (Lin e Chang [2001]), etc. O aparecimento de uma nova representação é justificado por melhorias na eficiência da avaliação das soluções, pelo tipo de esquemas de codificação, pela quantidade de redundância que existe na codificação ou pelo número total de configurações, relativamente às anteriormente existentes. De seguida, será dada uma breve visão histórica das diferentes representações que foram surgindo. Para a representação das relações de adjacência entre itens num padrão podem ser usados grafos de restrições (grafos acíclicos direcionados) [Fang e Wong, 1991]. Os grafos de restrições podem ser utilizados para representar padrões gerais. Um item é representado no grafo por um nó e uma aresta direcionada representa a relação de localização entre dois itens, figura 2.11.

Murata et al. [1995] propuseram uma representação que é considerada uma das mais marcantes de entre todas as representações, sendo ao longo dos tempos a mais utilizada pelos investigadores para representar padrões.

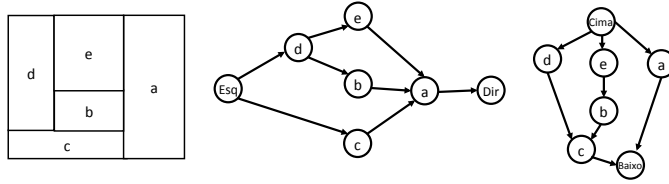


Figura 2.11: Grafos de restrição.

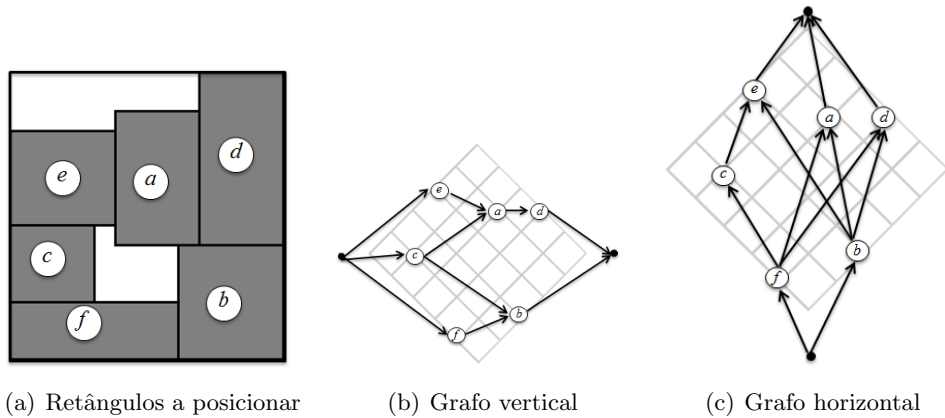


Figura 2.12: Representação SP (adaptado de Murata et al. [1995])

A ideia principal é usar um par de sequências para representar as posições relativas entre pares de itens, posicionar os itens numa grelha e, construir os respetivos grafos de restrições para avaliar o custo. Para cada par de itens (p.ex. I_1 e I_2) podem considerar-se quatro relações distintas, *esquerda-direita*, *direita-esquerda*, *cima-baixo* e *baixo-cima* correspondendo nas sequências, respetivamente, a $(I_1 I_2, I_1 I_2)$, $(I_2 I_1, I_2 I_1)$, $(I_1 I_2, I_2 I_1)$ e $(I_2 I_1, I_1 I_2)$. Da figura 2.12(b) e da figura 2.12(c) podem ser obtidas as seguintes sequências ($ecadfb, fcb ead$). Para aceder a diferentes vizinhanças, podem realizar-se as seguintes operações: rotação de um item, troca de dois itens numa sequência e troca de dois itens em ambas as sequências.

Outra representação apresentada pelos mesmos autores foi a *bounded sliceline gride*, Nakatake e Fujiyoshi [1997]. Esta representação consiste numa estrutura em grelha, sem dimensões físicas, para definir as relações ortogonais entre itens, figura 2.13. Apesar do objetivo ser a representação das posições relativas entre pares de itens, pode também ser usada como um algoritmo de compactação para determinar a posição exata dos itens. São

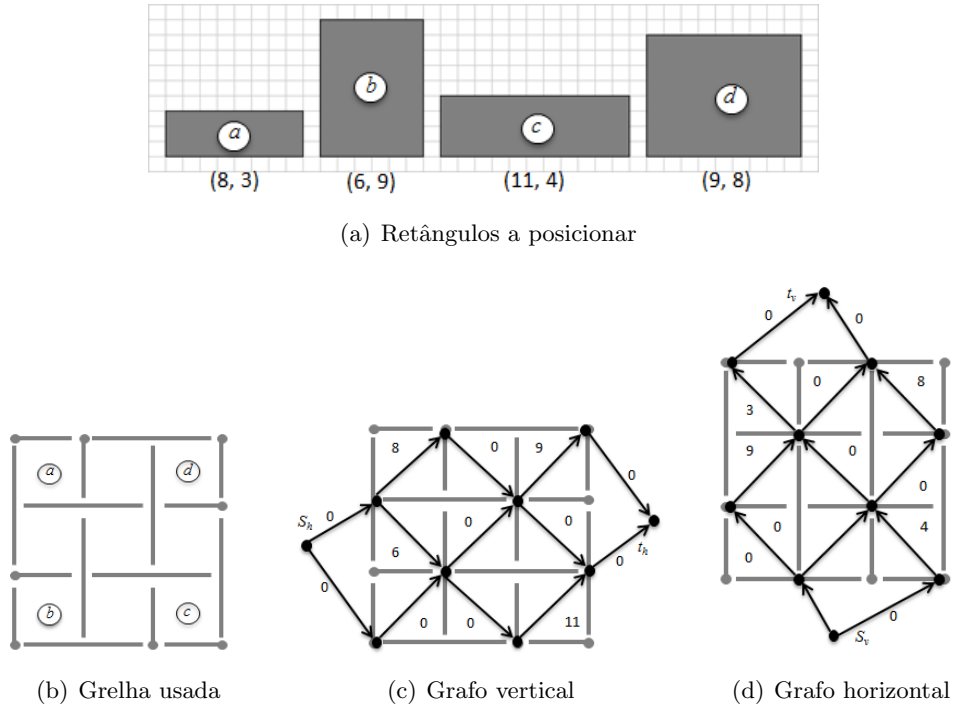


Figura 2.13: Representação BSG (adaptado de Nakatake e Fujiyoshi [1997])

usados grafos de restrições onde os vértices correspondem aos segmentos da grelha e onde as arestas direcionadas correspondem às posições relativas dos segmentos da grelha. As operações usadas nesta representação podem ser a troca de dois itens e a rotação de um item.

Uma representação baseada em árvores foi proposta por Yoshimura [1999], *O-tree*, com o objetivo de reduzir as redundâncias existentes nas anteriores representações. Nesta representação um posicionamento diz-se admissível se e só se nenhum item puder ser movido para a esquerda e para baixo relativamente a outros itens, isto é, todos os itens estão compactados na direção do eixo do x e do y . Nestas árvores, cada nó representa um item enquanto que uma aresta representa a relação de adjacência entre dois itens. Há dois tipos de *O-trees*: uma vertical e uma horizontal. O facto desta árvore possuir número arbitrário de ramos, para cada nó, torna as operações na árvore bastante complicadas sendo necessário o uso de uma codificação. Por exemplo, a *O-tree* representada na figura 2.14 pode ser codificada por duas sequências (001001101011000011110011, ahiljkbcdefg). Na primeira sequência um '0' indica que se está a percorrer a árvore descendendo o nó correspondente na

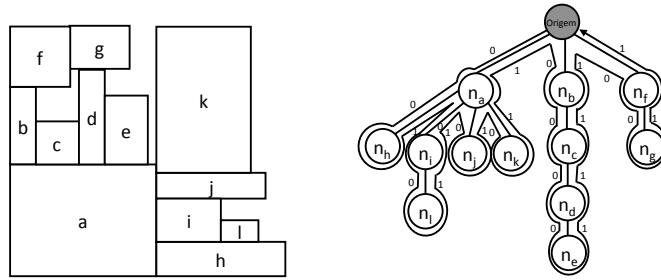


Figura 2.14: O-tree (adaptado de Yoshimura [1999]).

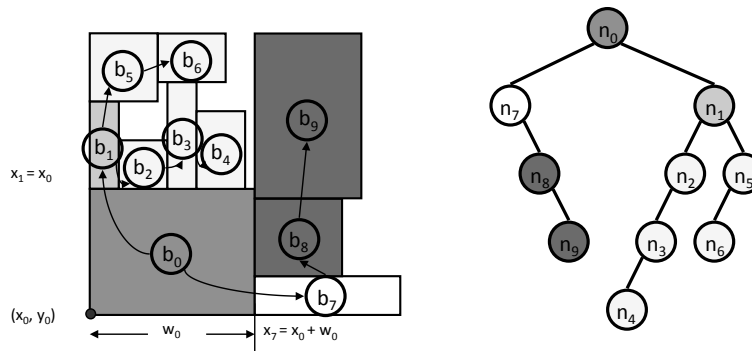


Figura 2.15: B*-tree (adaptado de Chang et al. [2000]).

árvore, enquanto um '1' indica que se está a percorrer de forma ascendente. A segunda sequência denota a ordem pela qual os itens são percorridos na árvore. Para a exploração do espaço de soluções pode eliminar-se um item da árvore e inseri-lo num nó externo com melhor custo.

Mais tarde, Chang et al. [2000] propuseram uma representação da solução baseada numa estrutura em árvore à qual chamaram B*-tree, figura 2.15. Uma B*-tree é uma árvore binária, que é construída recursivamente, em que cada nó representa um item. Cada item na árvore é posicionado no canto inferior esquerdo. Tal como a O-tree representa apenas padrões compactos. Esta estrutura em árvore contém duas subárvores que são construídas de modo independente. Nesta representação, pode usar-se como perturbações a rotação de um item, a troca de dois itens e a mudança de um item para outra posição.

Contrariamente ao que acontece com uma O-tree, numa B*-tree não é necessário construir grafos de restrições adicionais para a avaliação da função

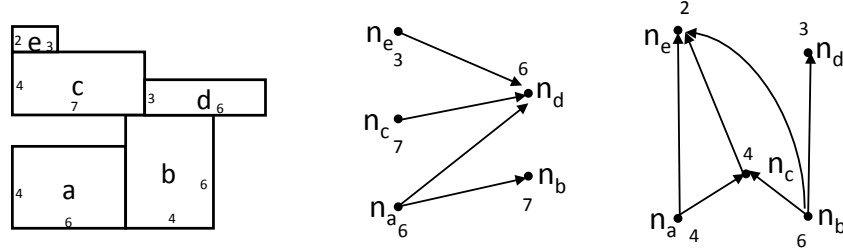


Figura 2.16: Posicionamento e correspondentes TCG (horizontal e vertical) (adaptado de Lin e Chang [2001]).

objetivo durante o posicionamento, tornando a fase de avaliação mais rápida.

Lin e Chang [2001] propuseram uma representação que tira partido das vantagens existentes nas representações *sequence pair*, *bounded sliceline grid* e *B*-tree*, designada por *transitive closure graph* (TCG). Esta representação descreve as relações geométricas entre itens baseadas em dois grafos (um horizontal e outro vertical), figura 2.16.

Tal como acontece com a *B*-tree* e, contrariamente ao que acontece com as representações SP, BSG, *O-tree*, a representação TCG não necessita de recorrer a grafos de restrições adicionais para a avaliação do custo durante o posicionamento, tornando esta avaliação mais rápida. Além disso, esta representação suporta atualizações incrementais durante as operações e mantém a informação dos itens na fronteira assim como a forma e a posição relativa dos itens. Uma das grandes vantagens, desta representação, reside no facto de que a relação entre os itens é transparente não só para a TCG mas também para as suas operações. Isto significa que o efeito de uma operação na mudança de relações geométricas entre pares de itens é conhecido antes do posicionamento, facilitando uma convergência mais rápida para a solução desejada. As perturbações permitidas são a rotação de um item, a troca de dois itens, a troca de uma aresta no grafo vertical e no grafo horizontal e a mudança de uma aresta do grafo vertical para o horizontal (ou vice-versa).

Para o PP de retângulos Imahori et al. [2003] propuseram três meta-heurísticas para custos espaciais gerais: uma pesquisa local com geração de várias soluções iniciais, uma pesquisa local iterativa e uma de pesquisa aleatória. Mais recentemente, os mesmos autores Imahori et al. [2005] desen-

volveram técnicas eficientes para avaliar soluções obtidas pela representação SP.

Sendo uma das representações mais usadas, a SP surge no trabalho de vários autores. Por exemplo, Drakidis et al. [2006] usara-na inserida numa abordagem baseada em algoritmos genéticos. O método usado é estendido para lidar com restrições de simetria. A representação SP surge também no trabalho de Chatterjee e Manikas [2007] conjuntamente com algoritmos genéticos. Neste trabalho pode encontrar-se um estudo comparativo da performance dos operadores e esquemas de seleção existentes nos algoritmos genéticos.

Uma abordagem híbrida foi proposta por Beltrán-Cano [2009]. Esta abordagem consiste em duas fases distintas. Na primeira fase fixa-se a largura e o PP é resolvido como *Strip Packing* usando um algoritmo híbrido entre os algoritmos *Greedy Randomized Adaptive Search Procedure* (GRASP) e *Variable Neighborhood Search* (VNS). Numa segunda fase é determinado um novo valor para a largura. No posicionamento é usado o conceito de contorno.

O PP de itens retangulares num objeto com dimensões desconhecidas é abordado no trabalho de Korf et al. [2008]. Usam uma pesquisa *branch-and-bound* através do espaço de atribuições parciais, assegurando que a área do retângulo exterior envolvente não excede a área mínima de qualquer solução encontrada até ao momento. Esta abordagem foi também usada por Onodera et al. [1991].

Recentemente, uma abordagem baseada em métodos de solução para o carregamento de contentores bidimensional e para o problema de *Strip Packing* foi proposta por Bortfeldt [2013] para o PP de minimização de área de itens retangulares. Trata-se de uma abordagem genérica de redução com a restrição de obter padrões guilhotináveis.

2.3.2 Problemas com Itens Ortogonais

Os trabalhos referidos anteriormente, para PP de minimização de área, apenas lidam com itens retangulares. No entanto, na literatura, as abordagens existentes para itens retangulares, têm vindo a ser estendidas para poderem lidar com itens com outras formas. Por exemplo, no posicionamento de itens em circuitos *VLSI*, polígonos ortogonais surgiram para um melhor aproveitamento da área do circuito integrado e para melhorar as ligações entre

itens, aumentando a performance do circuito integrado. As representações existentes para o posicionamento de retângulos foram sendo adaptadas de forma a permitir a sua aplicabilidade a problemas com polígonos ortogonais. Assim, é natural que a maioria das abordagens para PP com formas ortogonais usem as mesmas técnicas para problemas onde os itens são retangulares, dividindo os polígonos ortogonais em retângulos e impondo um conjunto de restrições necessárias para manter a forma original dos itens.

Um dos primeiros autores a proporem um algoritmo que lida com o problema de posicionamento com polígonos ortogonais foram Preas e VanCleave [1979]. Propuseram grafos para representar as relações topológicas (posições relativas) entre pares de itens. Um algoritmo iterativo de melhoramento foi usado para reduzir a área do retângulo exterior envolvente que contém todos os itens e as ligações entre itens. No entanto, este algoritmo apenas lida com instâncias de pequena dimensão.

Para encontrar o melhor posicionamento para polígonos ortogonais Lee [1993] propôs um algoritmo que se baseia num contorno que limita os itens. Cada item é representado por um conjunto de quatro perfis lineares que descrevem o contorno do item visto a partir dos quatro lados do mesmo. Esta abordagem é diferente da maioria das abordagens que lidam com polígonos ortogonais pois não divide os itens em retângulos. Este algoritmo apenas lida com instâncias de pequenas dimensões dada a complexidade de cálculo dos perfis.

Para lidar com itens em forma de L, Nakatake e Fujiyoshi [1997] propõem um método baseado na representação BSG. Cada item em forma de L é dividido em retângulos que são posicionados em compartimentos adjacentes. No entanto, tal como é referido em Fujiyoshi e Murata [2002], alguns padrões não podem ser gerados uma vez que não são representáveis todas as posições relativas entre pares de itens.

Mais tarde, a mesma representação é usada por Nakatake et al. [1998] para problemas onde os polígonos ortogonais têm restrições de posicionamento. Em primeiro lugar, lidam com os itens com posições fixas, onde é construída uma solução parcial e, de seguida, os polígonos ortogonais, divididos em retângulos, são incluídos no padrão. Os subitens (retângulos resultantes da divisão dos itens ortogonais) são posicionados um a um até que sejam posicionados nas posições relativas corretas. De seguida, os subitens são tratados como itens com posições fixas. No entanto, tal como acontece

com o trabalho anterior não é possível gerar todas as posições relativas entre pares de itens, Fujiyoshi e Murata [2000].

A maioria das abordagens adaptaram e estenderam o uso da representação SP para lidar com polígonos ortogonais. Kang e Dai [1998a] usaram essa representação para polígonos ortogonais convexos. Neste trabalho, tal como acontece na maioria dos trabalhos que usam SP, cada polígono ortogonal é dividido num conjunto de subitens retangulares e cada subitem é tratado na representação SP como um único item. Cada item resultante desta divisão é compactado na vertical e na horizontal e um processo *a posteriori* alinha as coordenadas x e y destes itens para produzir padrões ótimos resultantes da representação SP. Neste trabalho é apresentada uma prova de que existe sempre uma representação SP admissível para polígonos ortogonais convexos e vice-versa. Foram definidas três operações necessárias e suficientes para alterar uma representação SP admissível mantendo a representação resultante também admissível. No entanto, de acordo com Fujiyoshi e Murata [2002] os lemas e teoremas apresentados contêm alguns erros, tornando as provas insuficientes. Para o posicionamento de polígonos ortogonais convexos e côncavos Fujiyoshi e Murata [2000] também usaram esta representação. Apresentaram condições necessárias e suficientes para a geração de representações SP admissíveis no que diz respeito a grafos de restrições, não impondo quaisquer restrições às sequências. Apesar de não ser necessário nenhum processo para restaurar a forma original dos itens a pesquisa de representações SP admissíveis apenas depende da abordagem usada. No entanto, para alguns itens os grafos de restrição revelam-se muito complicados. Mais tarde, para lidar com itens em forma de L, esta representação foi usada por Fujiyoshi e Murata [2002]. Nesta abordagem, estes itens, analogamente a anteriores abordagens, são divididos em itens retangulares. Apresentaram provas de condições necessárias e suficientes, apenas dependentes da representação SP, para o restauro da forma original dos itens. Provaram que qualquer posicionamento é possível de ser representado e incluí a solução ótima. Um método novo que lida com polígonos ortogonais foi proposto por Tang e Wong [2004]. Usaram propriedades topológicas de representações SP admissíveis para eliminação de representações não admissíveis. Apenas são geradas representações SP que satisfazem essa propriedade.

Fazendo os itens pertencer a uma classe específica de polígonos ortogo-

nais convexos, Kang e Dai [1998b] propuseram um algoritmo baseado nas representações SP e BSG nas quais as restrições topológicas destes itens podem ser otimizadas. Estes itens são compactados através de restrições topológicas de forma independente nas direções x e y . Dado um posicionamento, as dimensões e formas dos itens são alteradas sendo todos compactados segundo as mesmas relações topológicas. Segundo Fujiyoshi e Murata [2002] este método é muito complicado e a topologia do posicionamento inicial não é representável.

Para lidar com itens ortogonais Pang et al. [2001] usaram a representação *O-tree*. Apesar de ter uma complexidade inferior, esta abordagem é menos flexível e mais restrita, uma vez que, cada item é dividido num conjunto de itens em forma de L e, no posicionamento, os itens têm que estar compactados.

Outra representação que foi adaptada para lidar com polígonos ortogonais foi a *B*-tree*, usada por Wu et al. [2000]. Tal como nas abordagens anteriores os itens são divididos em retângulos. Uma desvantagem desta representação resulta da rotação dos itens, sendo necessário uma nova repartição. Um processamento posterior é necessário para restaurar a forma inicial dos itens.

A representação TCG foi adaptada por Lin et al. [2002], onde também é necessário dividir os itens em retângulos. Apresentaram condições necessárias e suficientes para a obtenção de TCG admissíveis. Ao contrário do que acontece na maioria das abordagens, não necessitam de um processamento posterior para a obtenção da forma original dos itens uma vez que cada conjunto de subitens é tratado como um todo. Com este processo conseguem garantir a geração de TCG admissíveis quando é aplicada uma perturbação.

Para lidar com polígonos ortogonais Chen et al. [2010] propuseram um algoritmo baseado nos princípios de grau de cavidade e ocupação de cantos.

Chu e Young [2004] partem de um padrão e têm como objetivo alterar as formas e dimensões de alguns itens retangulares para polígonos ortogonais com o objetivo de preenchimento de espaços vazios, mantendo as áreas dos itens constantes e as posições relativas entre itens inalteradas (que descrevem através de grafos de restrição).

2.4 Problemas de Posicionamento Multiobjetivo de Retângulos

PPMO consistem em posicionar um conjunto de itens de forma a otimizar mais do que uma função objetivo. Por exemplo, quando se pretende minimizar o número de objetos disponíveis para posicionar um conjunto de itens e minimizar o número de padrões de corte; quando se pretende minimizar o desperdício resultante do posicionamento de itens e minimizar o custo do corte dos itens; outro exemplo surge quando se pretende maximizar a capacidade de um contentor com caixas de modo a satisfazer o maior número possível de clientes e minimizar a distância total do percurso que resulta da entrega das encomendas, etc. Dada a complexidade adicional dos PPMO, a maioria dos trabalhos lidam com os objetivos de forma sequencial, otimizando um de cada vez, ou resolvem-no como sendo mono-objetivo. De seguida apresentam-se algumas aplicações reais de PP com mais do que um objetivo. Na duas primeiras subseções serão descritos os problemas MO de posicionamento de módulos em circuitos integrados e no projeto físico de instalações. Na última subsecção serão apresentados outros PPMO que diferem dos anteriores nos objetivos considerados e na formas dos itens.

2.4.1 Posicionamento de Módulos em Circuitos VLSI

No PP de módulos em circuitos VLSI, figura 2.17, pretende-se, para além de reduzir o custo do material usado, reduzir o tamanho do circuito integrado, aumentar a performance (redução do atraso, redução do comprimento das ligações) e reduzir o tempo de fabrico para uma rápida entrada no mercado.

A maioria das publicações existentes para problemas MO, nesta aplicação real consideram para além da minimização da área a minimização da soma das distâncias que ligam um determinado conjunto de itens. Este problema de minimizar a área e a soma das distâncias foi tratado por Chen et al. [2009] que propõem uma abordagem baseada numa otimização por enxame de partículas (*particle swarm optimization*). Os itens disponíveis para posicionamento são retangulares. Na fase de posicionamento dos itens é apenas possível fazer uma estimativa para o valor da distância total, uma vez que nesta fase ainda não são conhecidas as posições exatas dos pinos de entrada e saída e dos pinos que ligam os itens entre si. Estes autores usam como estimativa da soma das distâncias o semi-perímetro da distância total. Con-

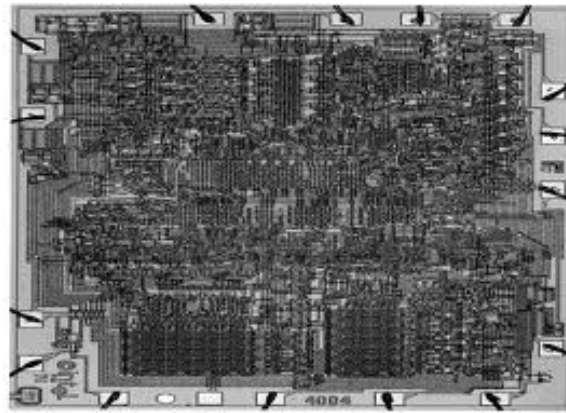


Figura 2.17: Circuito VLSI

sideram os dois objetivos através do método da soma ponderada, resultando num único objetivo. Estes autores referem que esta forma de medir a função objetivo é muito subjetiva, considerando que são objetivos difíceis de pesar porque têm propriedades diferentes. Chen et al. [2011] também usam soma ponderada para lidarem com os mesmos objetivos. A diferença desta abordagem para as demais é a existência de restrições de regularidade onde os itens são agrupados por tamanho e forma.

Para otimizar a minimização da área e a soma das distâncias, o algoritmo proposto por Dong et al. [2004] usa duas fases. Na primeira fase, usam programação quadrática para minimizar a distância total, o resultado deste processo iterativo é um padrão inicial com distância total mínima. Com base neste padrão, é estimada uma relação entre a largura e a altura do padrão que, sabendo o desperdício, pode ser calculada a área do padrão. Na segunda fase, um algoritmo baseado no primeiro é usado para melhorar o posicionamento tendo em conta a distância total calculada no passo anterior.

Para lidarem simultaneamente com a minimização da área e da distância total Fernando e Katkoori [2008] usam um algoritmo genético. É considerado o conceito de não dominância para a classificação das soluções obtidas. Usam dois operadores de cruzamento que constroem padrões através do uso de bons subpadrões.

Vários autores usaram abordagens baseadas em programação matemática usando diferentes técnicas. Para minimizar a área, a distância total e atrasos em circuitos, Sutanthavibul et al. [1991] usam programação inteira

mista e vão construindo o padrão adicionando itens a um conjunto de itens anteriormente posicionados. Para lidarem com o problema multiobjetivo usam uma combinação linear entre a área e a soma das distâncias.

Uma combinação linear entre a área e as distâncias é também usada por Kim e Kim [2003]. Estes autores combinam programação linear e arrefecimento simulado para o problema de minimização de área e distância total. Usam a representação SP para definirem as posições relativas entre pares de itens. Desenvolveram dois métodos para a obtenção do padrão que resulta da representação SP, um considerando um modelo de programação linear e outro através de um método construtivo. Estes dois métodos foram inseridos no algoritmo arrefecimento simulado para a obtenção de uma boa solução.

Para o problema de minimização de área e para garantir a existência de espaço para as ligações entre itens, Onodera et al. [1991] usam a técnica *branch-and-bound* para uma exploração efetiva de soluções, no entanto, esta abordagem apenas lida com instâncias de pequena dimensão. Para instâncias de grande dimensão fazem uma decomposição hierárquica do problema. A minimização da área e da distância total é realizada em dois passos. Num primeiro passo minimizam a área, sendo de seguida calculada a distância total. Referem que optaram por esta abordagem pois um aumento da área do padrão resulta num aumento da soma das distâncias. A minimização da área é decomposta em dois problemas de programação linear, minimizando a largura do padrão e a altura do padrão separadamente.

Para a minimização da área, minimizando simultaneamente a largura e a altura da área do posicionamento Watanabe e Hiroyasu [2004] aplicam algoritmos genéticos. Para representar as soluções usam a já referida representação SP.

2.4.2 Projeto Físico de Instalações

Neste problema considera-se um conjunto de instalações, que têm que ser colocadas dentro de um edifício, de modo a assegurar um arranjo que otimizava uma dada função objetivo. Neste problema podem ser considerados vários objetivos, Kusiak e Heragu [1987] tais como minimizar o custo de manipulação ou manobra de materiais, minimizar o uso de espaço em termos efetivos e económicos, minimizar o tempo global de produção, minimizar o custo de operação do equipamento e da planta. A distância entre unidades de produção é um fator que é usado para a avaliação de um determinado pa-

drão. Esta distância é, normalmente, medida considerando a distância entre o centro das instalações. O cálculo da soma das distâncias pode ser efetuado usando diferentes processos, sendo alguns apresentados na seção 3.5.

O problema de minimizar o custo de manobras de material, maximizar a proximidade entre departamento e minimizar o tempo de manuseio de materiais foi abordado por Singh e Singh [2010]. Formularam o problema colocando os objetivos numa única função através do uso de pesos. Como abordagem propõem uma heurística que funciona em três fases. Na primeira fase, é proposto um procedimento de normalização. Na segunda fase, são propostos quatro métodos diferentes para determinar os pesos relativos para cada função objetivo. Por último, na fase final são apresentados quatro padrões diferentes com base em quatro métodos.

O problema de minimizar o custo de manobras de material e maximizar a proximidade entre departamentos foi abordado por Ripon et al. [2010] que usam como abordagem de resolução algoritmos genéticos.

No problema abordado por Ye e Zhou [2007] são considerados dois objetivos que envolvem os custos de movimentação de materiais e exigências não-materiais relacionadas com a ideia de planejamento sistemático de padrões.

2.4.3 Outros Problemas de Posicionamento Multiobjetivo

Para além dos problemas referidos anteriormente surgem outros PP que abordam mais do que um objetivo. Serão de seguida apresentadas abordagens existentes para alguns PPMO que diferem dos anteriores nos objetivos e na forma dos itens.

O problema de *cutting stock* numa tira de comprimento infinito e largura fixa, considerando como objetivos a minimização do desperdício e a minimização do número de cortes necessários para o posicionamento, foi abordado por Coelho et al. [2012]. Usam a heurística GRASP para a geração de parte da população inicial de um algoritmo evolucionário multiobjetivo baseado em algoritmos genéticos. Para avaliar soluções usam quatro heurísticas de posicionamento diferentes: *First Fit*, *Best Fit*, *Next Fit* e uma variação da heurística *Next Fit*.

Outro problema multiobjetivo surge no trabalho de Geiger [2008] que propõe uma abordagem heurística ao problema de *Bin Packing* com dois objetivos: minimizar o número total de contentores e minimizar a média de

heterogeneidade das peças incluídas em cada contentor.

O problema biobjetivo de minimizar o custo e minimizar o número de grupos no posicionamento de itens com diferentes pesos em objetos foi tratado por Sathe et al. [2009]. Usam um algoritmo que combina uma estratégia de agrupamento hierárquico aglomerativo com um algoritmo que lida com problemas de agrupamento com muitas restrições.

Um exemplo de um PPMO surge no Problema de Seleção de Sortido de Objetos que consiste em determinar um conjunto de tamanhos de objetos a ter em *stock*. Este problema de seleção surge da dificuldade de ter muitos objetos disponíveis para posicionamento, ou por limitações de espaço, ou por custos inerentes ao armazenamento dos objetos, ou por demorar muito tempo a produzir os objetos. O objetivo passa por minimizar a soma de todos os custos inerentes ao processo. Para além deste objetivo existe um outro que surge quando é necessário posicionar um conjunto de itens nos objetos, sendo necessário minimizar o desperdício. Uma revisão bibliográfica de Problemas de Seleção de Sortido pode ser encontrada em Pentico [2008]. Este problema surge mal definido em alguns artigos, nomeadamente em Chen et al. [1993], Li e Chang [1998], Li e Tsai [2001], Li et al. [2002] e Lin [2006]. Estes autores definem o problema de Seleção de Objetos como um problema de minimização de área. O problema abordado é, na realidade o Problema de Seleção de Objetos, isto é, pretendem determinar um conjunto de objetos para existir em *stock*. O tamanho dos itens é baseado no historial de encomendas anteriores e o objetivo é determinar objetos de área mínima para o posicionamento dos itens. Este problema também pode ocorrer quando é fácil e rápida a produção do objeto, não sendo necessário o recurso a *stock*, sendo apenas produzido na altura em que os itens são necessários.

Um outro exemplo de um PPMO é apresentado no trabalho desenvolvido por Moreira [2001] que surge numa empresa metalomecânica situada em Portugal. Esta companhia produz e comercializa ferramentas de precisão de alta tecnologia de corte de madeira, plásticos e materiais compostos. O processo de produção inclui o corte de “cabeças” de ferramentas de corte (peças irregulares) que têm de ser cortadas de placas circulares (dispendiosas) feitas de tungsténio com uma fina camada de pó de diamante. Neste contexto surgem dois problemas de planeamento: a disposição de peças no padrão (problema de *nesting*), relacionado com o aproveitamento da placa

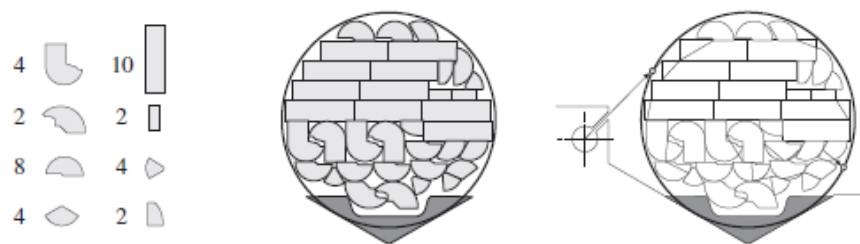


Figura 2.18: Maximização do aproveitamento e minimização do custo de corte (adaptado de Moreira [2001])

circular, e a definição do percurso de corte, relacionado com a minimização do custo do corte, figura 2.18. O posicionamento dos itens é feito usando um algoritmo de posicionamento baseado em trocas de peças em sequências. O procedimento de corte é executado de modo contínuo, i. e., o instrumento que corta nunca deixa a superfície a ser cortada. À medida que cada peça é totalmente cortada da chapa cai num recipiente especialmente colocado para o efeito. O grafo sobre o qual se utiliza a pesquisa do percurso de corte é dinâmico, alterando-se durante o próprio processo de corte e dependendo da ordem pelo qual os cortes são processados. Foi introduzido, pela primeira vez, o problema de carteiro rural dinâmico para a resolução deste tipo de problemas. Foram usadas, para a sua resolução, duas heurísticas construtivas: a heurística *higher up vertex* e a heurística *minimum empty path*. Este trabalho foi publicado mais tarde num artigo, Moreira et al. [2007], onde é apenas apresentado o procedimento de corte.

Imahori et al. [2008] resolvem o mesmo problema também em duas fases: empacotamento das peças irregulares numa placa circular e o corte dessas peças. As peças devem ser colocadas de modo a que o desperdício da placa seja mínimo. Na fase de corte, o objetivo é encontrar o percurso de corte ótimo, i.e., o comprimento total do percurso mínimo. Como a placa é de material duro a ferramenta de corte não se pode mover com rapidez logo o tempo de corte não deve ser negligenciado, sendo por isso importante encontrar o percurso mínimo. Estes autores dão importância à fase de corte, onde se pretende saber como cortar as peças quando é dado o posicionamento das peças na placa circular. Para resolver este problema foi introduzido um conceito novo chamado *force-support tree* onde um grande número de

restrições é representado de modo direto e conciso. Um caminho de corte que corta as peças dos nós das folhas dessa árvore em direção ao nó da raiz satisfaz a maior parte das restrições, podendo cada um concentrar-se nas restrições de ângulos abertos e na função objetivo de gerar o percurso de corte. O método utilizado encontrou bons caminhos muito rapidamente. No trabalho de Moreira (2001) o processo de corte é dinâmico contrariamente ao que acontece no trabalho dos últimos autores referidos em que o processo de corte é estático.

Em Moura e Oliveira [2008] é apresentado um *framework* para resolver problemas de determinação de rotas para veículos com janelas temporais e carregamento de contentores. Neste problema há um conjunto de clientes e uma frota de veículos heterogênea. Existe um armazém de onde saem os veículos para a visita aos clientes e para onde regressam no fim das entregas. Cada cliente tem uma encomenda para ser satisfeita, por um único veículo, e a janela temporal deve ser respeitada. As encomendas dos clientes devem ser respeitadas mesmo que tenha que ser usado outro veículo. A ordem do carregamento do veículo é inversa à ordem de visita dos clientes. A encomenda de cada cliente é posicionada toda junta no interior do veículo para aumentar a eficiência da operação de descarregamento. A capacidade do veículo em termos de peso, largura, altura e comprimento não deve ser excedida. Uma encomenda é constituída por um conjunto de caixas de diferentes tipos. O tipo de uma caixa é caracterizado pelas suas dimensões físicas, peso e orientação. Cada cliente é definido pelas coordenadas geográficas, janela temporal, encomenda (tipo de caixas e quantidades por tipo), peso total por encomenda e tempo de serviço. Há uma relação (dependência) entre o problema de determinação de rotas para veículos com janelas temporais e carregamento de contentores. A ideia principal é tentar servir com cada veículo o maior número possível de clientes e empacotar a sua encomenda de modo admissível ao mesmo tempo que se considera a minimização da distância total do percurso. São usados dois métodos diferentes de resolução. O primeiro método trata o problema de modo sequencial enquanto o segundo usa uma abordagem hierárquica. No primeiro, as rotas dos veículos e o carregamento dos contentores são planeados ao mesmo tempo. No segundo, numa primeira fase as rotas são construídas e só depois, para cada rota, os itens são posicionados nos contentores.

Capítulo 3

Modelos Matemáticos para Problemas de Posicionamento de Polígonos Ortogonais

É, tal como já foi referido, nos circuitos VLSI que o problema de posicionamento de polígonos ortogonais (PPPO) tem uma das maiores aplicações. Nesta aplicação real pretende-se, na maioria dos casos, minimizar a área do retângulo exterior envolvente e a soma das distâncias. A área do retângulo exterior envolvente que contém todos os itens denomina-se por área; e a soma das distâncias, ou apenas distâncias, é a soma do comprimento das ligações entre um determinado conjunto de itens. Estas ligações entre itens são denominadas nos circuitos por redes. Nos circuitos integrados torna-se necessário otimizar a área usada para o encaixe de mais funcionalidades no seu interior. Em relação à soma das distâncias é necessário um comprimento total das ligações mínimo para a redução de atrasos associados a fios longos e para acelerar as operações do circuito integrado. Uma vez que não existe um encaixe perfeito dos itens e, no posicionamento, surge algum desperdício (área que não está ocupada por nenhum item) existe um *trade-off* entre a minimização da área e a minimização da soma das distâncias.

Neste capítulo, um dos objetivos consiste no estudo de modelos matemáticos que existem para PP de minimização da área e, saber qual a sua capacidade para a resolução de instâncias de grande dimensão. Os mode-

los matemáticos existentes na literatura, para PP de minimização da área, apenas lidam com itens retangulares. Outro objetivo é adaptar os modelos existentes para que possam lidar com itens ortogonais. Em relação a PP de minimização da área do objeto e de minimização da soma das distâncias pretende-se estudar a componente multiobjetivo quando são considerados estes dois objetivos conjuntamente.

Nos PP há dois conjuntos de restrições comuns. Um dos conjuntos garante a não sobreposição dos itens e o outro conjunto garante o posicionamento dos itens no interior do objeto. As restrições de posicionamento dos itens no interior do objeto não são explicitamente consideradas no problema abordado nesta dissertação uma vez que, tal como já foi referido, o objeto não tem dimensões definidas, no entanto, surgem de forma implícita dada a necessidade de minimizar a área do retângulo envolvente. Os itens têm dimensões fixas, sendo necessário determinar as suas posições no plano, isto é, torna-se necessário determinar as coordenadas dos seus pontos de posicionamento (ordenada e abcissa dos centros dos itens).

Neste capítulo, apresentam-se abordagens baseadas em modelos matemáticos para PP quando se consideram como objetivos a minimização da área e/ou da soma das distâncias (soma do comprimento das ligações entre um determinado conjunto de itens). Na primeira secção, faz-se referência às restrições de não sobreposição existentes para PP. Na segunda secção, apresentam-se alguns exemplos de modelos onde o objeto ou objetos disponíveis para posicionar um conjunto de itens têm dimensões definidas. A terceira secção refere-se ao PP de minimização da área. Como a função objetivo é não linear, apresentam-se, nesta secção, dois conjuntos de aproximações usados na sua linearização. Num dos conjuntos a área é aproximada pelo perímetro e no outro conjunto a aproximação é baseada em funções definidas por ramos. Sendo a minimização da soma das distâncias outro objetivo que surge frequentemente em PP de itens, em circuitos VLSI, apresenta-se na última secção um modelo que considera a minimização da área e a minimização da soma das distâncias. Nesta secção serão também apresentadas algumas das estratégias usadas para a combinação destes dois objetivos.

Tabela 3.1: Variáveis de decisão e auxiliares

| | |
|--|--|
| x_i | abscissa do centro do item i |
| y_i | ordenada do centro do item i |
| r_{ij} | variável binária que indica se o item i está ou não à esquerda do item j |
| u_{ij} | variável binária que indica se o item i está ou não abaixo do item j |
| v_i | variável binária que indica se o item i rodou |
| w_i | largura do item i (variável auxiliar) |
| h_i | altura do item i (variável auxiliar) |
| $i, j \in \{1, 2, \dots, T_r\}$ e T_r é o número total de retângulos | |

Tabela 3.2: Parâmetros

| | |
|---|----------------------------------|
| M | constante suficientemente grande |
| d_{1i} | dimensão 1 do item i |
| d_{2i} | dimensão 2 do item i |
| $i \in \{1, 2, \dots, T_r\}$ e T_r é o número total de retângulos | |

3.1 Restrições de Não Sobreposição

Para a obtenção de soluções admissíveis torna-se necessário garantir que não há sobreposição entre os itens. Esta não sobreposição entre pares de itens é assegurada por um conjunto de restrições que podem ser divididas em dois grupos, um onde existem itens retangulares e outro onde existe pelo menos um item ortogonal. Por exemplo, para garantir que dois itens retangulares não se sobreponham é necessário assegurar que ocorre uma das seguintes situações: um dos itens está totalmente à direita do outro, totalmente à esquerda, totalmente acima ou totalmente abaixo. Estas relações são conseguidas com base na comparação de coordenadas. Estas condições originam um conjunto de disjunções de inequações, para cada par de itens. Para a modelização destas disjunções é necessário recorrer ao uso de variáveis binárias que crescem de forma polinomial com o aumento do número de itens. Para além destas variáveis binárias, como são permitidas rotações de itens, é necessário considerar outra variável binária. Sendo assim, é necessário utilizar as variáveis apresentadas na tabela 3.1 e os parâmetros da tabela 3.2.

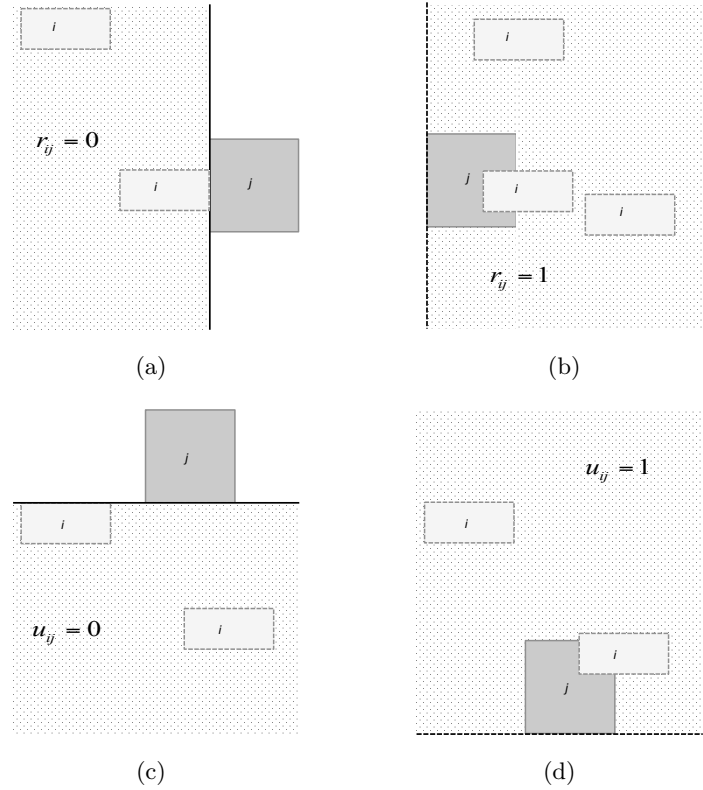


Figura 3.1: Valores das variáveis binárias r_{ij} e u_{ij} em função do posicionamento relativo entre dois itens i e j

A variável binária r_{ij} é igual a zero se o item i estiver completamente à esquerda do item j , figura 3.1(a); u_{ij} é igual a zero se o item i estiver completamente abaixo do item j , figura 3.1(c); estas variáveis são iguais a um, caso contrário, figuras 3.1(b) e 3.1(d). As posições relativas entre dois pares de itens i e j , e os respectivos valores das variáveis binárias r_{ij} e u_{ij} , estão representadas na figura 3.2. Por exemplo, na figura 3.2(b), $u_{ij} = 0$ significa que o item i tem que estar posicionado na região assinalada, onde a sua ordenada superior tem que ser menor ou igual à menor ordenada do item j ; como, nesta figura, $r_{ij} = 1$ a maior abcissa do item i não é menor do que a menor abcissa do item j ; a variável binária v_i é igual a zero se a peça não sofreu rotação e é igual a um, caso contrário; No caso de itens retangulares as rotações possíveis são 0 e 90° .

A componente geométrica, existente nos PP, consiste em garantir que os itens disponíveis para posicionamento não se sobrepõem. A não sobreposição

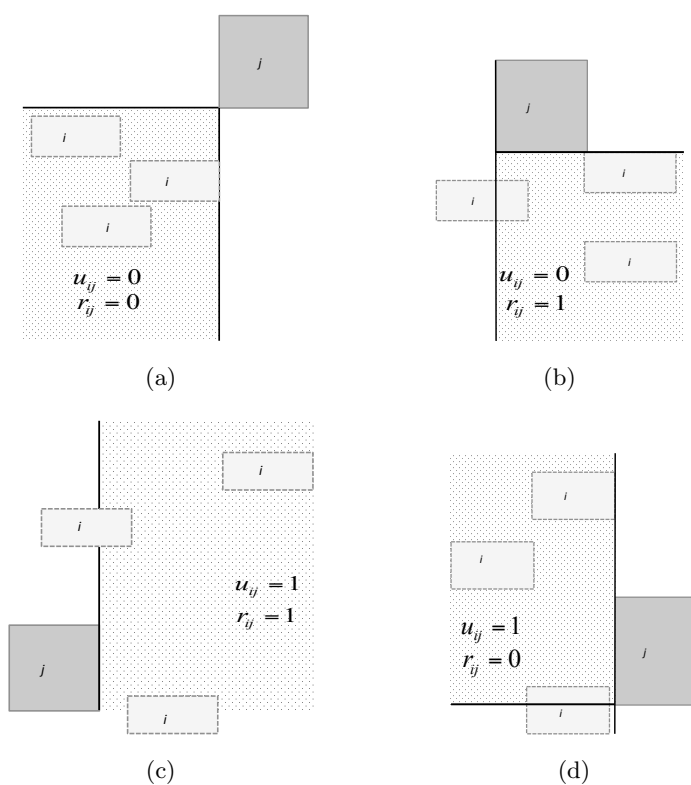


Figura 3.2: Relação entre as variáveis binárias r_{ij} e u_{ij} em função do posicionamento relativo entre dois itens i e j

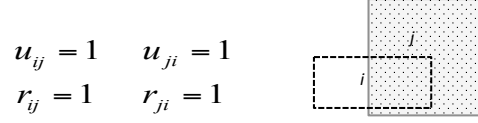


Figura 3.3: Valores das variáveis binárias r_{ij} e u_{ij} para uma situação de sobreposição entre dois itens i e j

de itens retangulares é assegurada pelas restrições apresentadas na equação (3.1).

$$\begin{aligned}
 x_i + 0.5w_i &\leq x_j - 0.5w_j + M \cdot r_{ij} & \forall i, j; \quad i \neq j; \\
 y_i + 0.5h_i &\leq y_j - 0.5h_j + M \cdot u_{ij} & \forall i, j; \quad i \neq j; \\
 r_{ij} + r_{ji} + u_{ij} + u_{ji} &\leq 3 & \forall i < j;
 \end{aligned} \tag{3.1}$$

(onde M é uma constante suficientemente grande)

As duas primeiras restrições da equação (3.1) garantem que os itens estão separados na direção horizontal e vertical, respetivamente. Quando as variáveis binárias u_{ij} , r_{ij} , u_{ji} e r_{ji} são simultaneamente iguais a 1 resulta numa sobreposição entre os itens i e j , figura 3.3, a não ocorrência desta situação é garantida pela terceira restrição da equação (3.1).

No problema proposto nesta dissertação, cada item i tem dimensões fixas que são conhecidas, d_{1i} e d_{2i} , e pode rodar. As restrições da equação (3.2), adicionadas ao modelo, permitem que os itens rodem.

$$\begin{aligned}
 w_i &= d_{1i} \cdot (1 - v_i) + d_{2i} \cdot v_i & \forall i; \\
 h_i &= d_{1i} \cdot v_i + d_{2i} \cdot (1 - v_i) & \forall i.
 \end{aligned} \tag{3.2}$$

Se no PP existe um conjunto de itens ortogonais para posicionamento, a forma mais direta para lidar com estes polígonos é fazer a sua divisão em retângulos, na figura 3.4 apresenta-se um exemplo desta divisão. Na tabela 3.3 apresentam-se os parâmetros e na tabela 3.4 as variáveis de decisão e auxiliares usadas na modelação deste problema.

Como os polígonos ortogonais são divididos em retângulos, as restrições apresentadas na equação (3.1) garantem que os retângulos que resultaram desta divisão não se sobrepõem. No entanto, como as rotações de itens ortogonais são quatro e as dos itens retangulares são apenas duas é necessário

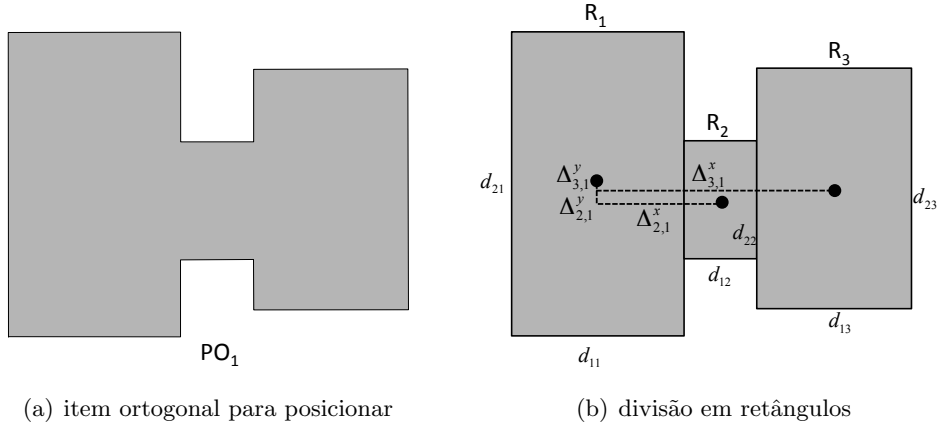


Figura 3.4: Divisão de um polígono ortogonal em retângulos

Tabela 3.3: Parâmetros relacionados com polígonos ortogonais

| | |
|---|---|
| F_b^x | distância, medida na horizontal, do centro do retângulo a ao centro do retângulo b antes de rodado o item ortogonal |
| F_b^y | distância, medida na vertical, do centro do retângulo a ao centro do retângulo b antes de rodado o item ortogonal |
| $a \in \{1, 2, \dots, T_o\}$ e T_o é o número total de polígonos ortogonais | |
| $b \in \{1, 2, \dots, R_o - 1\}$ e R_o é o número total de retângulos resultante da divisão do polígono ortogonal o | |

Tabela 3.4: Variáveis de decisão e auxiliares relacionadas com polígonos ortogonais

| | |
|----------------------------------|---|
| w_a | largura do item principal a de um polígono ortogonal |
| h_a | altura do item principal a de um polígono ortogonal |
| w_b | largura do item secundário b de um polígono ortogonal |
| h_b | altura do item secundário b de um polígono ortogonal |
| z_{ba}^k | variável binária que indica a rotação do item secundário b |
| Δ_{ba}^x | distância, medida na horizontal, entre o centro do retângulo a e o centro do retângulo b depois de rodado o item ortogonal |
| Δ_{ba}^y | distância, medida na vertical, entre o centro do retângulo a e o centro do retângulo b depois de rodado o item ortogonal |
| <hr/> | |
| $k \in \{1, 2, 3, 4\}$ | |
| $a \in \{1, 2, \dots, T_o\}$ | e T_o é o número total de polígonos ortogonais |
| $b \in \{1, 2, \dots, R_o - 1\}$ | e R_o é o número total de retângulos resultante da divisão do polígono ortogonal o |

adicionar ao modelo um conjunto de restrições que permita a rotação de itens ortogonais. Quando existe pelo menos um item ortogonal a posicionar, para além das restrições das equações (3.1) e (3.2), é necessário adicionar ao modelo restrições que garantam que as formas iniciais dos polígonos ortogonais não se alteram.

As variáveis de decisão consideradas são z_{ba}^k , onde $k \in \{1, 2, 3, 4\}$, $a \in \{1, 2, \dots, T_o\}$; T_o é o número total de polígonos ortogonais e $b \in \{1, 2, \dots, R_o - 1\}$; onde R_o é o número de retângulos resultante da divisão do polígono ortogonal o . Consideramos que um dos retângulos é o item principal e os restantes são os itens secundários, na figura 3.4(b) foi considerado principal o retângulo R_1 e secundários os retângulos R_2 e R_3 .

Tanto os itens secundários como o item principal, que constituem cada polígono ortogonal, podem rodar. Os itens ortogonais podem sofrer quatro rotações possíveis 0° , 90° , 180° e 270° . As equações (3.3) garantem que os itens ortogonais podem sofrer essas quatro rotações, sem que a forma original

dos itens se altere.

$$\begin{aligned}
w_b &= d_{1b} \cdot (z_{ba}^1 + z_{ba}^3) + d_{2b} \cdot (z_{ba}^2 + z_{ba}^4) \quad \forall a, b; \\
h_b &= d_{1b} \cdot (z_{ba}^2 + z_{ba}^4) + d_{2b} \cdot (z_{ba}^1 + z_{ba}^3) \quad \forall a, b; \\
w_a &= d_{1a} \cdot (z_{ba}^1 + z_{ba}^3) + d_{2a} \cdot (z_{ba}^2 + z_{ba}^4) \quad \forall a, b; \\
h_a &= d_{1a} \cdot (z_{ba}^2 + z_{ba}^4) + d_{2a} \cdot (z_{ba}^1 + z_{ba}^3) \quad \forall a, b; \\
z_{ba}^1 + z_{ba}^2 + z_{ba}^3 + z_{ba}^4 &= 1 \quad \forall a, b; \\
z_{ba}^k &= z_{b'a}^k \quad \forall k, a, b, b'; b' \neq b;
\end{aligned} \tag{3.3}$$

As duas primeiras restrições da equação (3.3) dizem respeito às rotações dos itens secundários e as duas seguintes às rotações do item principal. Um e um só dos z_{ba}^k pode ser igual a 1 e a rotação tem que ser a mesma quer para o item principal, quer para os itens secundários. Esta relação é garantida pela última restrição da equação (3.3).

Na figura 3.5 estão representadas as quatro orientações possíveis para polígonos ortogonais que dependem do valor das variáveis binárias $z_{ba}^1, z_{ba}^2, z_{ba}^3$ e z_{ba}^4 .

É necessário também adicionar ao modelo um conjunto de restrições auxiliares que permitam fixar as posições dos itens secundários em relação ao respetivo item principal. As variáveis auxiliares destas restrições são Δ_{ba}^x e Δ_{ba}^y , figura 3.5. As equações (3.4) permitem que a forma original dos itens ortogonais não se altere mesmo depois de rodados.

$$\begin{aligned}
\Delta_{ba}^x &= F_b^x \cdot z_{ba}^1 + F_b^y \cdot z_{ba}^2 - F_b^x \cdot z_{ba}^3 - F_b^y \cdot z_{ba}^4 \quad \forall a, b; \\
\Delta_{ba}^y &= F_b^y \cdot z_{ba}^1 - F_b^x \cdot z_{ba}^2 - F_b^y \cdot z_{ba}^3 + F_b^x \cdot z_{ba}^4 \quad \forall a, b.
\end{aligned} \tag{3.4}$$

Onde F_b^x e F_b^y são as distâncias do centro do item principal ao centro do item secundário, medidas na horizontal e na vertical, respetivamente. Assim, os centros dos itens secundários são definidos em relação ao item principal através das restrições apresentadas na equação (3.5).

$$\begin{aligned}
x_b &= x_a + \Delta_{ba}^x \quad \forall a, b; \\
y_b &= y_a + \Delta_{ba}^y \quad \forall a, b.
\end{aligned} \tag{3.5}$$

3.2 Posicionamento em Objetos Definidos

A título ilustrativo apresentam-se exemplos de modelização de objetivos em PP com objetos definidos. Um destes exemplos surge no Problema da

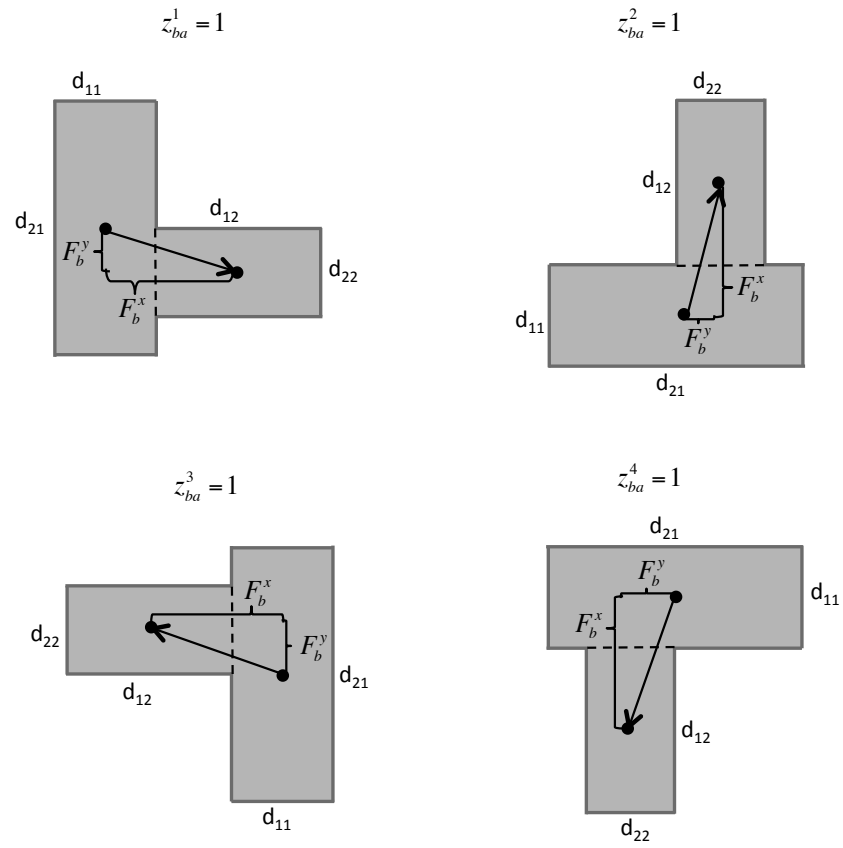


Figura 3.5: Rotações possíveis de itens ortogonais

Mochila onde está disponível um objeto, com comprimento W e altura H , para posicionar um conjunto de itens R_i com $i \in \{1, 2, \dots, n\}$, neste exemplo com formas retangulares, de largura w_i e altura h_i . O objetivo é maximizar o valor dos itens usados. Para modelizar este problema, para além das variáveis de decisão apresentadas na tabela 3.1 é necessário considerar uma variável binária, para cada item, que indica se o item R_i é posicionado no objeto. Assim, $z_i = 1$ se o item R_i for posicionado no objeto, e é igual a zero caso contrário. Considerando (W_e, H_e) e (W_d, H_d) as coordenadas do vértice inferior esquerdo e superior direito do objeto retangular, respetivamente. O Problema da Mochila pode ser modelizado através da função objetivo apresentada na equação (3.6), através das restrições de não sobreposição (equação (3.1)) e das restrições garantem que os retângulos são posicionados no interior do objeto (equação (3.7)).

$$\text{Max} \left(\sum_{i=1}^n w_i \cdot h_i \cdot z_i \right) \quad (3.6)$$

$$\begin{aligned} x_i - 0.5w_i &\geq W_e \quad \forall i; \\ x_i + 0.5w_i &\leq W_d \quad \forall i; \\ y_i - 0.5h_i &\geq H_e \quad \forall i; \\ y_i + 0.5h_i &\leq H_d \quad \forall i; \\ z_i &\in \{0, 1\} \end{aligned} \quad (3.7)$$

$$x_i, y_i, w_i, h_i, W_e, W_d, H_e, H_d \geq 0 \quad (3.8)$$

As restrições apresentadas na equação (3.7) ao contrário do que acontece com as restrições de não sobreposição, (equação (3.1)), não colocam dificuldades na resolução do modelo, uma vez que crescem linearmente com o número de itens.

Outro exemplo, de um PP num objeto definido é o problema de *Strip Packing* que é muito semelhante ao problema anterior, diferindo na função objetivo. Neste problema o objeto tem altura limitada e comprimento infinito pretendendo-se minimizar W , onde W é uma variável auxiliar que representa a largura do objeto. A segunda restrição da equação (3.7) necessita de ser adaptada para este problema, sendo alterada para a restrição apresentada na equação (3.9).

$$x_i + 0.5w_i \leq W. \quad (3.9)$$

O problema de *Bin Packing* é outro exemplo onde os objetos têm dimensões definidas. Neste problema há um conjunto de objetos disponíveis para posicionar um conjunto de itens. Por exemplo, na versão unidimensional deste problema, existe um conjunto de objetos O disponíveis para posicionar um conjunto de itens I . Cada objeto i tem uma capacidade c_i e cada item j tem um peso w_j . O objetivo é minimizar o número de objetos necessários para posicionar todos os itens. As variáveis de decisão são y_i que assume o valor 1 se o objeto i é usado e o valor 0 caso contrário e, x_{ij} é igual a 1 se o item j for colocado no objeto i e é igual a 0 caso contrário.

O modelo matemático para este problema pode ser obtido através da função objetivo apresentada na equação (3.10) e das restrições da equação (3.11).

$$\text{Min } \sum_{i \in O} y_i \quad (3.10)$$

$$\begin{aligned} \sum_{i \in O} x_{ij} &= 1 \quad \forall j \in I; \\ \sum_{j \in I} w_j x_{ij} &\leq c_i y_i \quad \forall i \in O; \\ x_{ij} &\in \{0, 1\} \quad \forall i \in O \quad \forall j \in I; \\ y_i &\in \{0, 1\} \quad \forall i \in O; \end{aligned} \quad (3.11)$$

A equação (3.10) indica que o objetivo é minimizar o número de objetos necessário para posicionar os itens. A primeira restrição da equação (3.11) assegura que cada item é posicionado num único objeto; a segunda restrição garante que não é excedida a capacidade de cada objeto.

3.3 Problemas de Minimização da Área do Objeto

Nos problemas de minimização da área, ao contrário do que acontece nos problemas apresentados na secção anterior, o objeto não tem qualquer dimensão definida, sendo o objetivo a minimização da sua área. Para além das variáveis de decisão apresentadas na tabela 3.1 e das restrições das equações (3.1)–(3.5) apresentadas na secção 3.1, que dizem respeito ao posicionamento dos itens, tenham eles formas ortogonais ou retangulares, é necessário a introdução de outras restrições e variáveis de decisão. As variáveis de decisão que é necessário introduzir no modelo, quando se pretende minimizar a área

Tabela 3.5: Variáveis de decisão auxiliares para a modelação da área

| | |
|-----|-------------------|
| W | largura do padrão |
| H | altura do padrão |

do retângulo exterior, dizem respeito às dimensões e coordenadas do padrão e estão apresentadas na tabela 3.5.

É aqui que surgem as dimensões do objeto, apesar de não serem consideradas de forma explícita. A primeira e segunda restrição da equação 3.12 permitem determinar a largura e a altura do padrão, respetivamente.

$$\begin{aligned} W &\geq (x_i + 0.5w_i) - (x_j - 0.5w_j) \quad \forall i, j; \\ H &\geq (y_i + 0.5h_i) - (y_j - 0.5h_j) \quad \forall i, j. \end{aligned} \quad (3.12)$$

O objetivo do PP de minimização da área do objeto pode ser formalmente definido pela expressão apresentada na equação (3.13).

$$\text{Min } (W \cdot H) \quad (3.13)$$

O termo $W \cdot H$ da função objetivo é não linear e não convexo. O facto de ser não linear impossibilita o uso de um *solver* para a resolução de modelos matemáticos lineares. Em muitas abordagens, o termo não linear da função objetivo é aproximado através de uma função linear, permitindo assim o uso de ferramentas disponíveis para modelos matemáticos lineares. Apesar de nos *solvers* não lineares, não existir a necessidade da linearidade da função objetivo, nem das restrições, exige-se que a função objetivo seja diferenciável e que as restrições formem um conjunto convexo. O grande problema, que dificulta a obtenção da solução ótima nos problemas de programação não linear, são os mínimos e máximos (extremos) locais da função objetivo. Sendo a função objetivo não convexa, dificulta o uso de ferramentas de programação não linear pela complexidade de convergir para a solução ótima. Por este motivo é natural o uso de aproximações em vez de recorrer a esta ferramenta de resolução do modelo.

3.3.1 Aproximações Lineares ao Cálculo da Área

Na função objetivo o termo $W \cdot H$ a ser minimizado é não linear. Em muitos trabalhos existentes na literatura este termo é linearizado. Nesta subsecção apresentam-se algumas aproximações usadas para esta linearização com base no semiperímetro do retângulo exterior envolvente.

Um processo para linearizar a área do retângulo exterior envolvente foi proposto por Sutanthavibul et al. [1991] onde supõem que uma das dimensões é conhecida, W , considerando como objetivo a minimização da outra dimensão, H . No entanto, sem considerarem uma restrição que controle a relação entre W e H obtêm-se muitas vezes padrões muito estreitos e altos. Este autores referem que, considerando esta relação, $a = \frac{W}{H}$, é possível descrever a área do padrão como função de uma única variável, $F = a.H^2$. O mínimo da função F , uma vez que $H > 0$, tem o mesmo valor de H que o mínimo da função $F_1 = H$. Estes autores incluem no modelo, para além das restrições de não sobreposição da equação (3.1), as restrições adicionais apresentadas na equação (3.14).

$$\begin{aligned} x_i + w_i &\leq a.H; \\ H &\geq y_i + h_i; \end{aligned} \tag{3.14}$$

sendo W e H a largura e a altura do padrão e x_i e y_i a ordenada e a abcissa do posicionamento do canto inferior esquerdo do item i , respetivamente.

A mesma estratégia de fixar uma das dimensões para linearizar o termo da função objetivo foi usada por Chen et al. [1993] e, mais recentemente, por Lim [2008].

Uma função logarítmica é proposta por Hakimi [1996] e Chen e Fan [1998] para a linearização da área. A expressão da área na função objetivo, equação (3.13), é aproximada por: $\overline{W} + \overline{H}$, considerando $W = e^{\overline{W}}$ e $H = e^{\overline{H}}$. Este tipo de abordagem é limitado, uma vez que apenas lidam com instâncias de pequena dimensão, dada a elevada complexidade para resolver problemas de programação convexa.

A expansão de Taylor é usada para a aproximação da área por Chen e Kuh [2000], tal como apresentado na equação (3.15).

$$\begin{aligned} f(W, H) &= W^0 \cdot H^0 + H^0(W - W^0) \\ &\quad + W^0(H - H^0) + (W - W^0)(H - H^0) \\ &\approx W^0 \cdot H^0 + H^0(W - W^0) + W^0(H - H^0) \end{aligned} \tag{3.15}$$

na equação (3.15), W^0 e H^0 são valores iniciais para a largura e altura do padrão, respetivamente. Os termos constantes apresentados na equação (3.15) são eliminados obtendo-se a equação (3.16).

$$\begin{aligned} \text{Min } f(W, H) &\approx \text{Min } \hat{f} \\ &= \text{Min } (H^0 \cdot W + W^0 \cdot H) \end{aligned} \quad (3.16)$$

Esta aproximação é tanto melhor quanto menor for o valor de $(W^* - W^0) \cdot (H^* - H^0)$, onde (W^*, H^*) é a solução ótima.

Uma das aproximações mais naturais para linearizar a área do retângulo exterior envolvente é o uso do perímetro. Esta abordagem foi usada por Yamazaki et al. [2000] para o posicionamento de polígonos convexos. Estes autores referem que esta aproximação reduz de forma significativa os cálculos uma vez que a formulação se reduz a programação linear. Usam como base o trabalho de Katoh e Ibaraki [1987], onde a programação quadrática foi implementada usando um problema paramétrico, isto é, consideram como objetivo minimizar a soma da largura do padrão com um escalar multiplicado pela altura do padrão (equação (3.17)) em vez da minimização da área (equação (3.13)).

$$\text{Min } (W + \lambda \cdot H) \quad (3.17)$$

Se λ for devidamente escolhido, a solução ótima da expressão apresentada na equação (3.17) é a solução ótima da apresentada na equação (3.13), onde λ é determinado através de uma pesquisa exaustiva no intervalo $[0, \infty[$.

Uma linearização para a minimização da área foi também apresentada por Zhao et al. [2006]. Estes autores consideram f_1 igual à área do padrão e f_2 a raiz quadrada dessa área. Naturalmente que minimizar f_1 é equivalente a minimizar f_2 . Além disso, f_2 é uma função convexa. Como as dimensões do padrão são positivas, o semiperímetro do padrão é sempre superior ou igual à raiz quadrada da área do padrão (equação (3.18)), sendo o semiperímetro do retângulo exterior envolvente usado para aproximar a sua área.

$$\frac{H + W}{2} \geq \sqrt{H \cdot W} \quad (3.18)$$

Usando, tal como em Chen e Kuh [2000], a expansão de Taylor (equação 3.16), resulta a equação (3.19). Da equação (3.18) resulta a equação (3.20). Assim, função f_3 (equação (3.21)) é usada para aproximar a área

(equação (3.13)).

$$\text{Min } f(W, H) = \text{Min } (H^0 \cdot W + W^0 \cdot H) \quad (3.19)$$

$$\begin{aligned} 0.5 \cdot H + 0.5 \cdot W &= \sqrt{H \cdot W} & W = H \\ \delta_1 \cdot H + \delta_2 \cdot W &= \sqrt{H \cdot W} & W \neq H \end{aligned} \quad (3.20)$$

$$f_3 = \delta_1 \cdot H + \delta_2 \cdot W \quad (3.21)$$

Obviamente que se pretende uma boa aproximação com um reduzido erro de estimativa da área assim, δ_1 e δ_2 devem ser contínuos onde a sua soma deve ser igual a um e, considerando $\tau = \frac{H}{W}$ obtém-se δ_1 e δ_2 onde $\tau \in [0.9, 1.1]$ tal como indicado na equação (3.22).

$$\begin{aligned} \delta_1 + \delta_2 &= 1; \\ \delta_1 &= \frac{1}{\sqrt{\tau} + 1}; \\ \delta_2 &= \frac{\sqrt{\tau}}{\sqrt{\tau} + 1}. \end{aligned} \quad (3.22)$$

3.3.2 Aproximações ao Cálculo da Área baseadas em Funções Definidas por Ramos

Nesta subsecção apresenta-se um processo diferente de aproximação do termo não linear da função objetivo com recurso a funções definidas por ramos. Em algumas das aproximações que se seguem, de forma a simplificar as expressões que serão apresentadas, usa-se x para designar W (largura do padrão) e y para H (altura do padrão).

Uma aproximação baseada numa função definida por ramos foi proposta por Li e Chang [1998] para PP de Seleção de Objetos. Para a linearização do termo $x \cdot y$ que aparece na função objetivo, consideram x e y variáveis delimitadas, onde $0 \leq x \leq \bar{x}$ e $0 \leq y \leq \bar{y}$, com \bar{x} e \bar{y} constantes, onde x e y são representadas pelas equações (3.23)–(3.24). Nestas equações ε_x e ε_y são números positivos pequenos, $\bar{\varepsilon}_x$ e $\bar{\varepsilon}_y$ são constantes pré-especificadas que são os limites superiores de ε_x e ε_y , respetivamente. θ e δ são variáveis binárias e G e H são números inteiros que representam o número de variáveis binárias necessárias para representar x e y . Das equações (3.23)–(3.24) pode obter-se

a equação (3.25) multiplicando x por y .

$$x = \bar{\varepsilon}_x \sum_{g=1}^G 2^{g-1} \theta_g + \varepsilon_x; \quad (3.23)$$

$$y = \bar{\varepsilon}_y \sum_{h=1}^H 2^{h-1} \delta_h + \varepsilon_y; \quad (3.24)$$

$$\begin{aligned} xy &= \left(\bar{\varepsilon}_x \sum_{g=1}^G 2^{g-1} \theta_g y + \varepsilon_x \right) y \\ &= \bar{\varepsilon}_x \sum_{g=1}^G 2^{g-1} \theta_g y + \varepsilon_x \bar{\varepsilon}_y \sum_{h=1}^H 2^{h-1} \delta_h + \varepsilon_x \varepsilon_y; \end{aligned} \quad (3.25)$$

Os termos $\theta_g y$ e $\delta_h \varepsilon_x$ da equação (3.25) têm que ser linearizados. Essa linearização é feita com base na proposição 1, também apresentada em Li e Chang [1998]).

Proposição 1 *O problema de otimização de $\text{Min } \theta y$, com $\theta \in (0, 1)$, $0 \leq y \leq \bar{y}$, $y \in F$, F conjunto de soluções admissíveis, pode ser linearizado por:*

$$\text{Min } y \quad (3.26)$$

Sujeito a:

$$\begin{aligned} y + \bar{y}(\theta - 1) &\leq z; \\ z &\geq 0; \\ \theta &\in (0, 1); \\ y &\in F \end{aligned} \quad (3.27)$$

□

Logo o termo $x.y$ da função objetivo pode ser substituído pela expressão linearizada apresentada na equação (3.28). O modelo proposto por Li e Chang [1998] é constituído pela função objetivo (equação (3.28)), pelas restrições de não sobreposição (equação (3.1)) e pelas restrições adicionais apresentadas na equação (3.29).

$$\text{Min } \left(\bar{\varepsilon}_x \sum_{g=1}^G 2^{g-1} z_g + \bar{\varepsilon}_y \sum_{h=1}^H 2^{h-1} u_h \right) \quad (3.28)$$

$$y + \bar{y}(\theta_g - 1) \leq z_g, \quad g = 1, 2, \dots, G; \quad (3.29)$$

$$z_g \geq 0;$$

$$\varepsilon_x + \bar{\varepsilon}_x(\delta_h - 1) \leq u_h, \quad h = 1, 2, \dots, H;$$

$$u_h \geq 0;$$

$$\theta_g, \delta_h \in (0, 1).$$

Tal como é referido em Li et al. [2002], uma das desvantagens do modelo proposto por Li e Chang [1998] é o número de variáveis binárias adicionais que é necessário para a linearização da função objetivo ($G + H$ variáveis adicionais) e, quanto menor forem os erros ε_x e ε_y , maior é o tamanho de G e H , sendo necessário um tempo computacional mais elevado para resolver o modelo. Para colmatar este elevado número de variáveis binárias, Li e Tsai [2001] propõem uma linearização diferente para a área. Em primeiro lugar resolvem o modelo constituído pela função objetivo definida pela expressão apresentada na equação (3.30), pelas restrições de não sobreposição (equação (3.1)) e pela restrição adicional que impõe que a largura do padrão seja maior do que a respetiva altura (equação (3.31)), e obtêm uma solução denotada por z_0 .

$$\text{Min } z = \text{Min } (W + H) \quad (3.30)$$

$$W \geq H \quad (3.31)$$

De seguida resolvem o modelo onde o objetivo é maximizar a largura do padrão (equação (3.32)), onde às restrições de não sobreposição (equação (3.1)) se adiciona uma restrição que assegura que a largura do padrão é maior ou igual que a altura e a soma destas duas dimensões é igual à solução z_0 (equação (3.33)). Esta solução z_0 foi obtida pelo modelo anteriormente apresentado.

$$\text{Max } W \quad (3.32)$$

$$W \geq H \quad (3.33)$$

$$W + H = z^0.$$

Naturalmente esta abordagem só pode ser usada para instâncias de pequena dimensão, pois envolve a resolução de dois modelos distintos. Segundo

os resultados apresentados pelos autores só é viável resolver instâncias até nove itens.

Outra estratégia para linearizar o termo quadrático é aplicar a função logaritmo à área do padrão. Um exemplo desta situação é apresentada por Li et al. [2002]. Estes autores, tal como em Li e Chang [1998], impõem um limite inferior e superior à largura e à altura admissíveis para o padrão (equação (3.34)). A função logarítmica é aproximada por uma função definida por ramos, onde cada ramo é um segmento de reta, figura 3.6. O termo quadrático da função objetivo (equação (3.35)) é aproximado pela expressão apresentada na equação (3.36) que é aproximada pela função definida por ramos da equação (3.37).

$$K = \{0 < \underline{x} \leq x \leq \bar{x}, 0 < \underline{y} \leq y \leq \bar{y}, x, y \in F\} \quad (3.34)$$

$$\begin{aligned} &\text{Min } xy \\ &x, y \in K; \end{aligned} \quad (3.35)$$

$$\text{Min } (\ln(x) + \ln(y)) \quad (3.36)$$

Assim, o modelo proposto consiste em minimizar a expressão apresentada na equação (3.37), pelas restrições que asseguram que os itens não se sobrepõem (equação 3.1) e por dois grupos de restrições adicionais, equação (3.38) e equação (3.39), respetivamente. O grupo 1 e o grupo 2 de restrições referem-se à linearização de $\ln(x)$ e $\ln(y)$, respetivamente.

$$\begin{aligned} \text{Min } &\left(\ln(a_1) + s_1(x - a_1) + \sum_{j=2}^{m_r-1} (s_j - s_{j-1})(a_j u_j + x - a_j - w_j) + \right. \\ &\left. + \ln(b_1) + t_1(x - b_1) + \sum_{j=2}^{n_r-1} (t_j - t_{j-1})(b_j v_j + y - b_j - q_j) \right) \end{aligned} \quad (3.37)$$

Grupo 1:

$$\begin{aligned} &-\bar{x}u_j \leq x - a_j \leq \bar{x}(1 - u_j); \\ &-\bar{x}u_j \leq w_j \leq \bar{x}u_j; \\ &-\bar{x}(u_j - 1) + x \leq w_j \leq \bar{x}(1 - u_j) + x, \quad u_j \geq u_{j-1}; \\ &\text{para } j = 2, 3, \dots, m_r \text{ (variável } x); \\ &a_1, a_2, \dots, a_{m_r} \in S_r \text{ conjunto dos pontos de quebra de } \ln(x); \\ &a_1 = \underline{x} < a_2 < \dots < a_{m_r} = \bar{x}; \end{aligned} \quad (3.38)$$

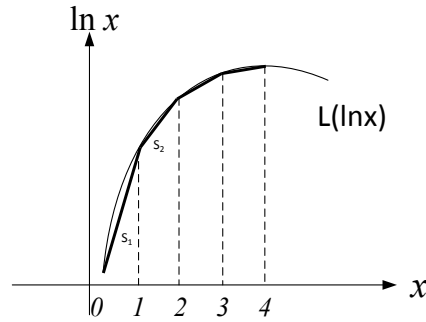


Figura 3.6: Aproximação por ramos (adaptado de Li et al. [2002])

Grupo 2:

$$\begin{aligned}
 & -\bar{y}v_j \leq y - b_j \leq \bar{y}(1 - v_j); \\
 & -\bar{y}v_j \leq q_j \leq \bar{y}v_j; \\
 & -\bar{y}(v_j - 1) + y \leq q_j \leq \bar{y}(1 - v_j) + y, \quad v_j \geq v_{j-1}; \\
 & \text{para } j = 2, 3, \dots, n_r \text{ (variável } y); \\
 & b_1, b_2, \dots, b_{n_r} \in T_r \text{ conjunto dos pontos de quebra de } \ln(y); \\
 & b_1 = \underline{y} < b_2 < \dots < b_{n_r} = \bar{y};
 \end{aligned} \tag{3.39}$$

onde u_j e v_j são variáveis binárias e $w_j, q_j \geq 0$.

Os testes computacionais realizados por Li e Chang [1998], Li e Tsai [2001] e Li et al. [2002] sugerem que a diversidade das dimensões dos retângulos tem influência nos resultados. As instâncias usadas têm quatro, cinco, oito e nove retângulos. Nas duas maiores instâncias, existem apenas três tipos de itens que são quadrados. Neste tipo de instâncias o resultado é um padrão onde os itens se ajustam facilmente.

3.4 Modelo Matemático Desenvolvido

Uma vez que o modelo matemático para o PPPO de minimização da área foi apresentado nas secções anteriores parcelarmente, será apresentado, nesta secção, o modelo completo de forma a facilitar a sua compreensão. Na tabela 3.6 apresentam-se as variáveis de decisão e auxiliares usadas na modelação do PPPO de minimização da área, enquanto que na tabela 3.7 se apresentam os parâmetros. Nas equações (3.40)-(3.48) é apresentada a função objetivo e as restrições usadas.

Tabela 3.6: Variáveis de decisão e auxiliares (modelo completo)

| | |
|---|--|
| x_i | abscissa do centro do item i |
| y_i | ordenada do centro do item i |
| r_{ij} | variável binária que indica se o item i está ou não à esquerda do j |
| u_{ij} | variável binária que indica se o item i está ou não abaixo do j |
| v_i | variável binária que indica se o item i rodou |
| z_{ba}^k | variáveis binárias que indicam a rotação do polígono ortogonal i |
| w_i | largura do item i |
| h_i | altura do item i |
| W | largura do padrão |
| H | altura do padrão |
| w_a | largura do retângulo principal a de um polígono ortogonal |
| h_a | altura do retângulo principal a de um polígono ortogonal |
| w_b | largura do retângulo secundário b de um polígono ortogonal |
| h_b | altura do retângulo secundário b de um polígono ortogonal |
| Δ_{ba}^x | distância, medida na horizontal, entre o centro do retângulo a e o centro do retângulo b depois de rodado o item ortogonal |
| Δ_{ba}^y | distância, medida na vertical, entre o centro do retângulo a e o centro do retângulo b depois de rodado o item ortogonal |
| <hr/> | |
| $i, j \in \{1, 2, \dots, T_r\}$ e T_r é o número total de retângulos | |
| $k \in \{1, 2, 3, 4\}$ | |
| $a \in \{1, 2, \dots, T_o\}$ e T_o é o número total de polígonos ortogonais | |
| $b \in \{1, 2, \dots, R_o - 1\}$ e R_o é o número total de retângulos resultante da divisão do polígono ortogonal o | |

Tabela 3.7: Parâmetros (modelo completo)

| | |
|----------|---|
| M | constante suficientemente grande |
| d_{1i} | dimensão 1 do item i |
| d_{2i} | dimensão 2 do item i |
| F_b^x | distância, medida na horizontal, entre o centro do retângulo a e o centro do retângulo b antes de rodado o item ortogonal |
| F_b^y | distância, medida na vertical, entre o centro do retângulo a e o centro do retângulo b antes de rodado o item ortogonal |

$$\text{Min } (W + H) \quad (3.40)$$

Sujeito a:

$$\begin{aligned} x_i + 0.5w_i &\leq x_j - 0.5w_j + M \cdot r_{ij} \quad \forall i, j \quad i \neq j \\ y_i + 0.5h_i &\leq y_j - 0.5h_j + M \cdot u_{ij} \quad \forall i, j \quad i \neq j \\ r_{ij} + r_{ji} + u_{ij} + u_{ji} &\leq 3 \quad \forall i < j \end{aligned} \quad (3.41)$$

$$\begin{aligned} w_i &= d_{1i} \cdot (1 - v_i) + d_{2i} \cdot v_i \quad \forall i \\ h_i &= d_{1i} \cdot v_i + d_{2i} \cdot (1 - v_i) \quad \forall i \end{aligned} \quad (3.42)$$

$$\begin{aligned} w_a &= d_{1a} \cdot (z_{ba}^1 + z_{ba}^3) + d_{2a} \cdot (z_{ba}^2 + z_{ba}^4) \quad \forall a, b \\ h_a &= d_{1a} \cdot (z_{ba}^2 + z_{ba}^4) + d_{2a} \cdot (z_{ba}^1 + z_{ba}^3) \quad \forall a, b \end{aligned} \quad (3.43)$$

$$\begin{aligned} w_b &= d_{1b} \cdot (z_{ba}^1 + z_{ba}^3) + d_{2b} \cdot (z_{ba}^2 + z_{ba}^4) \quad \forall a, b \\ h_b &= d_{1b} \cdot (z_{ba}^2 + z_{ba}^4) + d_{2b} \cdot (z_{ba}^1 + z_{ba}^3) \quad \forall a, b \end{aligned} \quad (3.44)$$

$$\begin{aligned} z_{ba}^1 + z_{ba}^2 + z_{ba}^3 + z_{ba}^4 &= 1 \quad \forall a, b \\ z_{ba}^k &= z_{b'a}^k \quad \forall k, a, b, b'; b \neq b' \end{aligned} \quad (3.45)$$

$$\begin{aligned} \Delta_{ba}^x &= F_b^x \cdot z_{ba}^1 + F_b^y \cdot z_{ba}^2 - F_b^x \cdot z_{ba}^3 - F_b^y \cdot z_{ba}^4 \\ \Delta_{ba}^y &= F_b^y \cdot z_{ba}^1 - F_b^x \cdot z_{ba}^2 - F_b^y \cdot z_{ba}^3 + F_b^x \cdot z_{ba}^4 \quad \forall a, b \end{aligned} \quad (3.46)$$

$$\begin{aligned} x_b &= x_a + \Delta_{ba}^x \quad \forall a, b \\ y_b &= y_a + \Delta_{ba}^y \quad \forall a, b \end{aligned} \quad (3.47)$$

$$\begin{aligned} W &\geq (x_i + 0.5w_i) - (x_j - 0.5w_j) \quad \forall i, j \\ H &\geq (y_i + 0.5h_i) - (y_j - 0.5h_j) \quad \forall i, j \end{aligned} \quad (3.48)$$

$$\begin{aligned} v_i, r_{ij}, u_{ij}, z_{ba}^1, z_{ba}^2, z_{ba}^3, z_{ba}^4 &\in \{0, 1\} \\ w_i, h_i, w_a, h_a, w_b, h_b, x_i, y_i, W, H &\geq 0 \end{aligned}$$

As restrições da equação (3.41) e da equação (3.42) garantem que não há sobreposições entre pares de itens e que os itens podem rodar (0 ou 90^0), respectivamente. Se no conjunto dos itens a posicionar existirem polígonos ortogonais é feita uma divisão em itens retangulares, onde um dos retângulos é o principal (a) e os restantes são secundários (b). As restrições da equação (3.43) e da equação (3.44) adicionadas ao modelo garantem que os polígonos ortogonais podem rodar 0 , 90^0 , 180^0 ou 270^0 . No entanto, após a rotação, é necessário garantir que a forma inicial dos polígonos ortogonais é mantida o que é assegurado pelas equações (3.46) e (3.47). Finalmente, as restrições da equação (3.48) permitem determinar o espaço ocupado pelos itens e, consequentemente, a determinação das dimensões do padrão.

3.5 Problema Integrado de Minimização da Área e Distâncias

É, principalmente nos circuitos VLSI que surgem associados os objetivos de minimizar a área do retângulo exterior envolvente e de minimizar a soma das distâncias. Idealmente, pretende-se que ambos sejam otimizados simultaneamente uma vez que os resultados de um têm influência no outro. Dada a complexidade do PP, com os objetivos de minimizar a área do retângulo exterior envolvente e minimizar a soma das distâncias resultantes das ligações entre os itens, muitas abordagens dividem o problema, resolvendo-o de forma sequencial. Numa primeira fase são determinadas as posições relativas entre pares de itens de modo a minimizar a distância total. Na fase seguinte são determinadas as posições exatas dos itens, mantendo inalteradas as posições relativas da fase anterior, de modo a minimizar a área. Apesar desta abordagem sequencial ter a vantagem de poder lidar com instâncias de grande dimensão, uma vez que os dois passos são realizados de forma relativamente independente, a qualidade da solução não é tão boa como em abordagens que resolvem o problema de forma integrada.

Nesta secção é apresentada uma formulação matemática para o problema da minimização da área e minimização da soma das distâncias. As distâncias são denominadas, nos circuitos integrados, por comprimento total do fio gasto para ligar os itens pertencentes a uma mesma rede. Para este problema é dado um conjunto de itens que pode ser constituído por retângulos e/ou polígonos ortogonais. A informação acerca das ligações entre peças é dada

na forma de *netlist*, onde para cada rede são indicadas as peças que lhe pertencem. Uma rede é um conjunto de itens ligados entre si.

Os problemas que contêm mais do que um objetivo podem ser resolvidos por diversos métodos com características diferentes, não existindo um método melhor do que todos os outros. Em Clímaco et al. [2003] pode encontrar-se uma descrição de vários métodos existentes para a resolução de problemas MO. A escolha do método varia com o que se pretende obter. Pode-se querer encontrar a fronteira eficiente ou apenas uma solução de acordo com a importância que cada objetivo tem para o tomador de decisão.

Um dos métodos mais comuns para resolver problemas MO é a combinação dos objetivos numa única função objetivo usando um vetor de pesos. Desta forma, o problema original é transformado num problema com um único objetivo com as restrições originais. Este método pode ser usado para obter uma aproximação da fronteira eficiente, resolvendo o modelo para diferentes valores do vetor dos pesos. Contrariamente ao que acontece se o vetor dos pesos for fornecido pelo tomador de decisão, onde se obtém uma única solução para o problema que é ótima para os valores dos pesos fornecidos. Apesar da simplicidade ser uma das vantagens deste método não há, no entanto, quando se pretende gerar a fronteira eficiente, garantia de uma boa cobertura, isto é, não se sabe se as soluções encontradas estão bem distribuídas entre todas as soluções do problema. Outra desvantagem resulta no tempo computacional que tem que se dispor para gerar diferentes soluções, principalmente em problemas onde a geração de uma solução é por si só computacionalmente dispendiosa. Outro método existente para a resolução de problemas MO consiste em otimizar uma das funções objetivo, transformando as restantes, $p - 1$, em restrições que são delimitadas por um vetor $\epsilon \in \mathbb{R}^{p-1}$. Este método é conhecido por ϵ -*constraint*. A função objetivo a ser otimizada é escolhida *a priori*, bem como o vetor ϵ . Neste método, para que existam soluções admissíveis, é necessário que este vetor seja escolhido de forma adequada, isto é, a sua escolha deve garantir a existência de um ponto que satisfaça as restrições do problema. A vantagem deste método surge no controle da cobertura da fronteira eficiente mas, tal como no método anterior, o problema tem que ser resolvido várias vezes para encontrar um conjunto de soluções eficientes. Um terceiro método consiste no uso do conceito de distância entre dois pontos para a determinação de uma solução eficiente. Esta distância pode ser calculada usando vários tipos de métri-

Tabela 3.8: Variáveis de decisão relacionadas com as distâncias

| | |
|---------------|---|
| w_k^r | largura do retângulo envolvente da rede k |
| h_k^r | altura do retângulo envolvente da rede k |
| x_k^{r-esq} | abscissa esquerda da rede k |
| x_k^{r-dir} | abscissa direita da rede k |
| y_k^{r-inf} | ordenada inferior da rede k |
| y_k^{r-sup} | ordenada superior da rede k |

cas. O que se pretende com este método é a determinação da solução mais próxima a um determinado ponto, denominado ponto de referência. Este ponto de referência pode ser o ponto ideal. Define-se ponto ideal aquele que é composto pelos máximos individuais de cada uma das funções objetivo.

O modelo, baseado no primeiro método apresentado, que combina os objetivos de minimização da área, A , e minimização da soma das distâncias, D , através de um vetor de pesos pode ser formalmente definido pela função objetivo apresentada na equação (3.49), pelas restrições das equações (3.1)–(3.5), apresentadas na secção 3.1, e pelas restrições apresentadas na equação (3.51) que dizem respeito ao cálculo das distâncias.

$$\text{Min } ((1 - \alpha) \cdot D + \alpha \cdot A) \quad (3.49)$$

$$\begin{aligned} x_k^{r-esq} &\leq x_i^{centro} \quad \forall i, k; i \in E_k; \\ x_k^{r-dir} &\geq x_i^{centro} \quad \forall i, k; i \in E_k; \\ y_k^{r-inf} &\leq y_i^{centro} \quad \forall i, k; i \in E_k; \\ y_k^{r-sup} &\geq y_i^{centro} \quad \forall i, k; i \in E_k; \\ x_k^{r-dir} - x_k^{r-esq} &= w_k^r \quad \forall k; \\ y_k^{r-sup} - y_k^{r-inf} &= h_k^r \quad \forall k; \end{aligned} \quad (3.50)$$

onde E_k é o conjunto de itens incluídos na rede k , com $D = \sum_{i,j \in R} c_{ij} \cdot [(x_i - x_j)^2 + (y_i - y_j)^2 - d_{ij}^2]^2$ e $A = W \cdot H$. As variáveis c_{ij} e d_{ij} representam a ligação e a distância entre o item i e o item j , respetivamente. As variáveis de decisão são, para além das mencionadas na tabela 3.1, secção 3.1, e das apresentadas na tabela 3.5 secção 3.3, as apresentadas na tabela 3.8.

O objetivo é minimizar a distância total entre itens que estão ligados entre si e a área do retângulo exterior envolvente que contém todos os itens,

sem sobreposições. Tal como já foi referido anteriormente, o termo da função objetivo referente à área não é linear. O mesmo acontece com o termo relativo à soma das distâncias, sendo necessário usar uma aproximação deste termo para que se possa usar a mesma abordagem usada para a minimização da área. De seguida são apresentadas algumas das aproximações usadas para essa linearização.

3.5.1 Aproximações ao Cálculo de Distâncias

A maioria dos trabalhos, na área do posicionamento de itens em circuitos VLSI, que têm como objetivos a minimização da área e da distância total usam estimativas para a minimização da distância total. A construção de um circuito, dada a sua complexidade, é feita de forma sequencial. A distância total exata não pode ser determinada sem se saber a posição exata dos pinos de entrada e saída e dos pinos dos itens. Nos circuitos, na fase de posicionamento, os itens são como caixas pretas, sem qualquer informação acerca do posicionamento exato destes pinos. Por este motivo, torna-se necessário o recurso a estimativas do comprimento total do fio. Na fase de posicionamento dos itens é apenas tida em consideração uma aproximação desta medida que será posteriormente otimizada na fase do *detailed routing*. Naturalmente, uma rápida estimativa tem uma elevada influência no desempenho do algoritmo de posicionamento.

Uma das métricas mais usadas, no posicionamento de itens em circuitos integrados, para medir o comprimento total que liga dois itens, é a distância de *Manhattan*. Esta métrica usa apenas retas verticais e horizontais, figura 3.7. Pode ser definida pela equação (3.51).

$$D(a, b) = \sum_{k=1}^2 |a_k - b_k| \quad (3.51)$$

Os métodos mais usados para estimar o comprimento do fio são:

- Semiperímetro do retângulo exterior envolvente do centro de todos os itens que estão ligados entre si. Este cálculo é repetido para todas as redes. A distância total é a soma de todos os cálculos parciais. É uma das estimativas mais utilizadas pois é calculada rapidamente e, para ligações de 2 ou 3 itens, é uma aproximação exata (a maioria dos circuitos tem este número de itens ligados), figura 3.8(a).

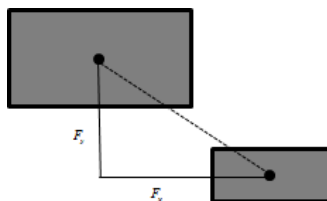


Figura 3.7: Distância de Manhattan = $F_x + F_y$

- Grafo completo - onde é calculada a distância entre todos os pares de itens. A distância pode ser a Euclidiana ou de *Manhattan*. Para uma rede com n itens, um grafo completo consiste em $\frac{n(n-1)}{2}$ arestas. Uma vez que cada árvore tem $n - 1$ arestas, que é $\frac{2}{n}$ vezes o número de arestas no grafo completo, a estimativa do comprimento da árvore, usando este método é dado por:

$$C = \frac{2}{n} \sum (\text{dist. pares de itens que pertencem à mesma rede}),$$

representado na figura 3.8(b).

- Origem para destino - onde a saída de cada item é ligada a todas as entradas dos outros itens da rede por ligações separadas. Esta estimativa calcula a distância da saída para as entradas. Apesar de ser uma estimativa rápida de calcular, não é uma boa estimativa para a distância por fazer uma diferenciação entre a entrada e a saída. Na maioria dos algoritmos essa diferenciação não é feita, figura 3.8(c).
- *Steiner Tree* - é o caminho mais curto formado por segmentos apenas com ângulos retos que liga um conjunto de itens. Neste método, um fio pode ramificar desde cada ponto ao longo do seu comprimento para ligar a outro item pertencente à mesma rede. O problema de encontrar a menor *Steiner Tree* é *NP*-completo, figura 3.8(d).
- Menor *Spanning Tree* - é uma árvore que une todos os itens que pertencem a uma mesma rede. Ao contrário do que acontece com a *Steiner Tree* uma ligação apenas pode ramificar a partir da localização dos itens. Para n itens pertencentes a uma rede, a árvore pode ser construída determinando as distâncias entre pares de itens, e ligar as menores $n - 1$ arestas que não formam um ciclo. O algoritmo de Kruskal calcula a menor *Spanning Tree* em tempo polinomial, figura 3.8(e).

- Menor cadeia - é uma sequência de retas que parte de um item para o mais próximo, que liga ao mais próximo, e assim sucessivamente, até que todos os itens estejam ligados, figura 3.8(f).

De seguida serão referidos alguns trabalhos que usam estimativas para o cálculo das distâncias. O PP com o objetivo de minimizar a soma das distância foi tratado por Tang et al. [2006]. Estes autores usam uma abordagem eficiente de custo mínimo baseada em fluxo. Esta abordagem garante obter a distância total, em tempo polinomial. É dada uma determinada topologia (posição relativa entre pares de itens) e, através da distribuição do desperdício, é obtida a distância total mínima. Muitas abordagens, apesar de obterem padrões com área mínima são sub-ótimos para os outros objetivos. Estes autores fazem notar que a distribuição do desperdício tem influência no valor da minimização da distância total, i.e, padrões que têm a mesma área e a mesma posição relativa entre itens podem ter diferentes distâncias totais apenas com uma mudança das posições do desperdício, tal como se mostra na figura 3.9. Para a estimativa da distância total usam o semiperímetro do retângulo exterior envolvente que contém os centros de todos os itens que estão ligados entre si. Estes autores referem que podem ser facilmente adicionadas restrições e ser usada uma função objetivo composta.

O PP com o objetivo de minimizar a distância total através da distribuição do desperdício foi também abordado por Li et al. [2011] através de uma abordagem baseada numa heurística construtiva. Estes autores referem que o trabalho de Tang et al. [2006] apresenta vários erros: em primeiro lugar partes do semiperímetro original não são consideradas no gráfico de rede; não são definidas quaisquer arestas que tenham em consideração o desperdício entre itens; o semiperímetro é fixo, com ou sem, o algoritmo de fluxo de custo mínimo e, referem que os resultados obtidos por Tang et al. [2006] não são fiáveis.

No trabalho de Caldwell et al. [1999] são distinguidos três tipos de estimativa para a distância:

- Estimativa *a priori*, isto é, procura-se estimar a distância total de um padrão antes do posicionamento dos itens. Em circuitos, esta estimativa pode ser útil para obter medidas aproximadas disponíveis para as ligações entre itens, para medir o desempenho do circuito, etc.;
- Estimativa *a posteriori* ocorre quando é dado um posicionamento fixo

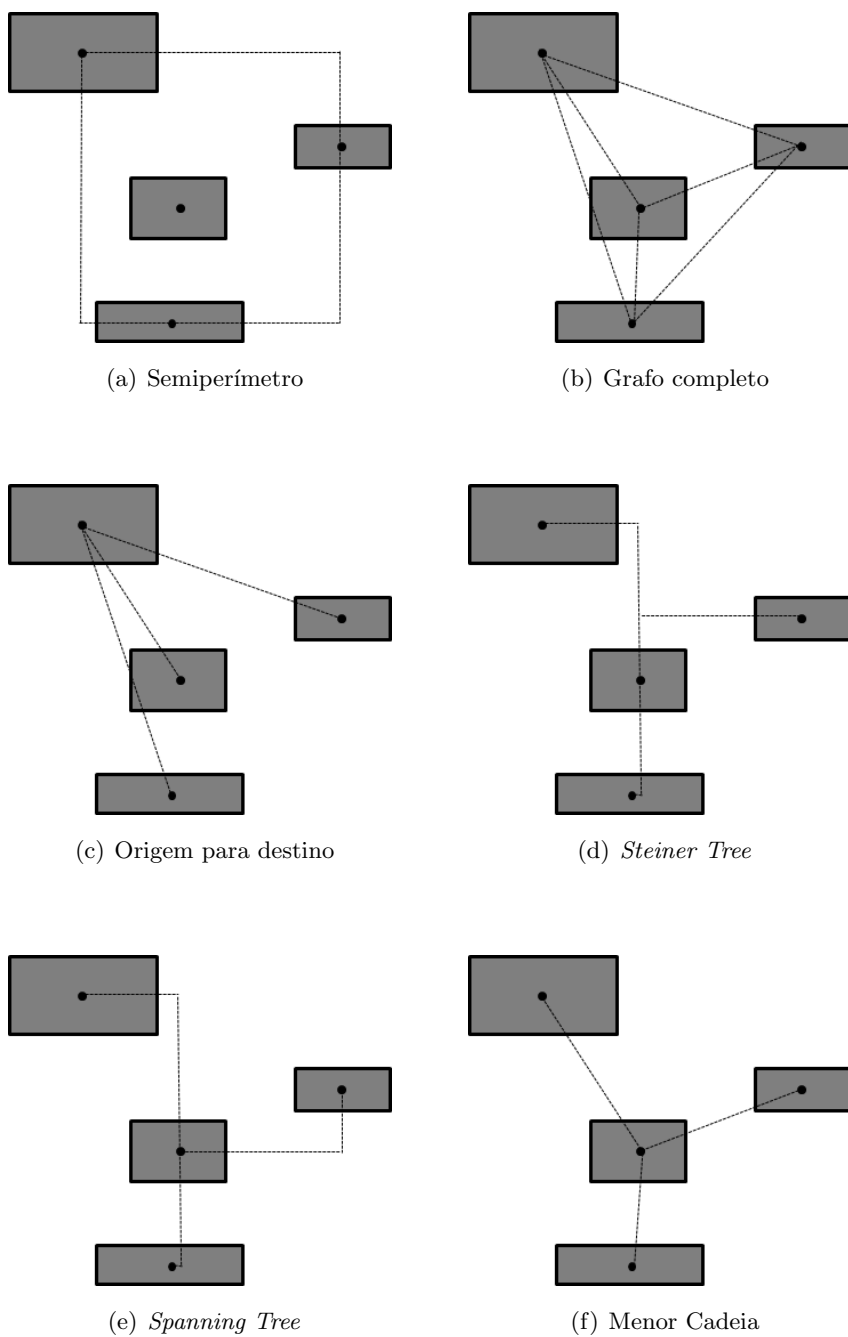


Figura 3.8: Diferentes formas de calcular as distâncias

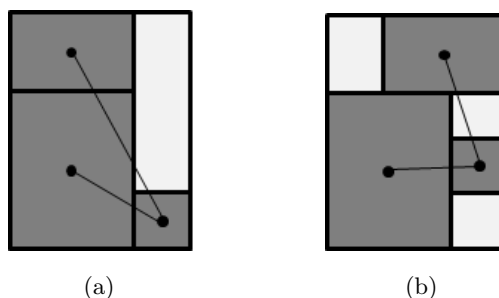


Figura 3.9: Influência do desperdício na minimização das distâncias (adaptado de Tang et al. [2006])

e se quer estimar o comprimento do fio depois do roteamento;

- Estimativa *on-line* acontece quando se pretende estimar a distância total durante o posicionamento. Esta estimativa pode ser usada, por exemplo, para terminar o posicionamento, antes de serem colocados todos os itens, porque se torna óbvio que irá resultar numa má solução.

Nesta dissertação usou-se, como estimativa das distâncias, o semiperímetro do menor retângulo envolvente que contém os centros de todos os itens que estão ligados entre si. A soma das distâncias é determinada pela soma de todos os semiperímetros.

3.5.2 Aproximações à Minimização da Área e Distâncias

A maioria das abordagens que usam modelos matemáticos que consideram o objetivo de minimizar a área e de minimizar a soma das distâncias lidam com estes dois objetivos combinando-os de forma linear, isto é, multiplicando cada um deles por um coeficiente e somando os dois termos deste produto. Os coeficientes são escolhidos arbitrariamente produzindo soluções, não dominadas, diferentes para diferentes coeficientes. Adya e Markov [2003] fazem notar que a área e a distância têm dimensões diferentes. A distância entre itens é da mesma ordem de grandeza que as dimensões x e y do padrão enquanto que a área tende a ter magnitude bastante maior do que o comprimento do fio, podendo tomar-se a raiz quadrada da área para que os termos fiquem com a mesma magnitude. Estes autores referem também que quando se pretende relacionar a área de um padrão com o seu perímetro, deve notar-se que é dependente da relação que existe entre

a largura e a altura do padrão. Por exemplo, o retângulo com largura 1 e altura 1000 tem perímetro igual a 2002, o retângulo com largura 10 e altura 100 tem perímetro igual a 220 e ambos têm área igual a 1000. Uma vez que a relação largura/altura muda durante o posicionamento não é correto considerar um coeficiente fixo. Além disso, a soma das distâncias, em geral, é muito maior do que o perímetro, dado o grande número de ligações entre itens. Este autores realçam que, das experiências que realizaram, salvo para coeficientes particulares, a área domina a soma das distâncias. Torna-se, portanto, muito difícil, quando se consideram estes dois objetivos, obter um método que explore apenas soluções não dominadas. Jagannathan et al. [2002] mostram que, com uma escolha apropriada dos coeficientes, qualquer ponto que pertence ao conjunto de pontos não dominados pode ser obtido. Adya e Markov [2003] também usam uma combinação entre os dois objetivos através de coeficientes. Para o PP de itens com os objetivos de minimizar a área e a soma das distâncias Kim e Kim [2003] usam programação inteira mista. O termo não-linear, referente à área, que aparece na equação (3.13), é linearizado aproximando-o através do semiperímetro do retângulo exterior envolvente, isto é, $(W \cdot H)$ é aproximado por $(W + H)$. Para o cálculo da soma das distâncias usam a aproximação, que é usualmente considerada pela maioria dos autores, do semiperímetro do menor retângulo exterior envolvente que contém os centros das peças que estão ligadas.

A função objetivo que combina a área e a distância apresentada na equação (3.52) é alterada para a equação (3.53).

$$\text{Min} \left((1 - \alpha) \cdot \sum_{k=1}^{nRedes} (w_k^r + h_k^r) + \alpha \cdot (W \cdot H) \right) \quad (3.52)$$

$$\text{Min} \left((1 - \alpha) \cdot \sum_{k=1}^{nRedes} (w_k^r + h_k^r) + \beta \cdot (W + H) \right) \quad (3.53)$$

onde α e β são constantes que determinam a importância relativa entre a soma das distâncias e a área e a soma das distâncias e o perímetro, respectivamente. O valor de β deve ser determinado de modo a que o semiperímetro (equação (3.54)) seja uma boa aproximação da área (equação (3.55)).

$$\beta \cdot (W + H) \quad (3.54)$$

$$\alpha \cdot (W \cdot H). \quad (3.55)$$

Kim e Kim [2003] consideram β igual a α multiplicado pela raiz quadrada da soma das áreas dos itens (A) dividida por dois, (equação (3.56)).

$$\beta = \alpha \frac{\sqrt{A}}{2} \quad (3.56)$$

Esta aproximação só é boa se W e H tiverem medidas próximas, sendo necessário introduzir uma restrição que controle a relação entre a largura e a altura. Considerando que a altura e a largura do padrão são iguais à raiz quadrada de A (equação (3.57)), a área do padrão é igual a A e o semiperímetro é igual a duas vezes a raiz de A (equação (3.58)).

$$W = H = \sqrt{A} \quad (3.57)$$

$$W \cdot H = A \quad (3.58)$$

$$W + H = 2\sqrt{A},$$

como se pretende que o segundo termo da equação (3.52) seja uma boa aproximação do segundo termo da equação (3.53), igualando estes dois termos (equação (3.59)) e resolvendo em ordem a β (equação (3.60)) obtém-se a equação (3.61) que resulta de substituir os valores da equação (3.58) na equação (3.60).

$$\beta(W + H) = \alpha(W \cdot H), \quad (3.59)$$

$$\beta = \alpha \frac{W \cdot H}{W + H}, \quad (3.60)$$

$$\beta = \alpha \left(\frac{A}{2\sqrt{A}} \right) = \alpha \frac{\sqrt{A}}{2}. \quad (3.61)$$

Assim, com as aproximações apresentadas anteriormente, nesta dissertação optou-se por aproximar a área através do semiperímetro do retângulo exterior envolvente que contém todos os itens e a soma das distâncias através do semiperímetro do retângulo exterior envolvente que contém os centros de todos os itens que estão ligados entre si. Os dois objetivos foram combinados numa única função objetivo usando um vetor de pesos. Assim, a função objetivo anteriormente apresentada na equação (3.49) foi substituída pela expressão apresentada na equação (3.62).

$$\text{Min} \left((1 - \alpha) \cdot \sum_{k=1}^{nRedes} (w_k^r + h_k^r) + \alpha \cdot (W + H) \right) \quad (3.62)$$

3.6 Experiências Computacionais

Nesta secção, realizam-se algumas experiências computacionais com os modelos matemáticos, apresentados neste capítulo, de minimização da área do objeto e da minimização da área e da soma das distâncias. Estas experiências têm como objetivo avaliar as capacidades dos modelos em várias situações. Num primeiro conjunto de experiências, para o modelo matemático de minimização da área do objeto, pretendeu-se saber quantos itens (apenas retangulares) é possível considerar em simultâneo de modo a que o modelo seja resolúvel em tempo útil, permitindo avaliar o limite em função do número total de itens. Num segundo conjunto de experiências pretendeu-se mostrar que o modelo de minimização da área pode ser facilmente estendido a itens ortogonais e, tal como no conjunto de experiências anterior, avaliar o limite em função do número total de itens que podem ser ortogonais ou retangulares. Finalmente, num terceiro conjunto de experiências, são considerados dois objetivos, a minimização da área e a minimização da soma das distâncias. Pretende-se estudar, com este conjunto de experiências, se estes dois objetivos são concordantes ou antagónicos, isto é, pretende-se saber se quando se minimiza um dos objetivos o outro também é minimizado.

Para o primeiro conjunto de experiências, com itens retangulares, começou-se com uma instância com cinco retângulos e foi-se sucessivamente adicionando um novo retângulo até se atingir um determinado tempo limite. Estas instâncias foram obtidas a partir da instância AMI33 (com origem no projeto de circuitos VLSI) com as dimensões dos retângulos são todas diferentes. Os resultados deste primeiro conjunto de experiências estão disponíveis na tabela 3.9. Nesta tabela, apresenta-se o número total de itens (retângulos) e o número de variáveis binárias do modelo para cada instância. Nesta tabela apresenta-se também o valor da função objetivo, o *gap* e o tempo de cálculo. Os resultados desta tabela mostram, tal como era esperado, o impacto que o número de itens tem no tempo de resolução e no número de variáveis binárias. Pode verificar-se que a simples adição de um item resulta num grande aumento no número de variáveis binárias e num acréscimo enorme no tempo de resolução. Como consequência, na resolução dos modelos com dez e onze itens retangulares foi atingido o tempo limite (3600s), não tendo sido possível obter a solução ótima. A solução ótima para a instância com sete retângulos apresenta-se na figura 3.10.

Para as experiências realizadas com instâncias que contêm itens ortogo-

Tabela 3.9: Resultados das experiências computacionais com o modelo matemático de minimização da área do objeto para itens retangulares

| # Itens | # Ret. | # Var. bin. | Obj. | gap (%) | Tempo (s] |
|---------|--------|-------------|------|---------|-----------|
| 5 | 5 | 44 | 483 | — | 0.23 |
| 6 | 6 | 65 | 560 | — | 1.44 |
| 7 | 7 | 90 | 623 | 0.0004 | 7.74 |
| 8 | 8 | 119 | 679 | 0.0098 | 192.43 |
| 9 | 9 | 152 | 735 | 0.0100 | 3032.08 |
| 10 | 10 | 189 | 133 | 16.814 | TL |
| 11 | 11 | 231 | 1225 | 2.2857 | TL |

TL (tempo limite)= 3600s

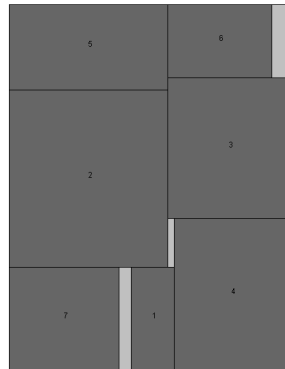


Figura 3.10: Exemplo de um padrão com 7 retângulos

Tabela 3.10: Resultados das experiências computacionais com o modelo matemático de minimização da área do objeto para itens ortogonais

| # Itens | # Ret. | # Ort. | # Var. bin. | Obj. | gap (%) | Tempo (s) |
|---------|--------|--------|-------------|------|---------|-----------|
| 5 | 9 | 4 | 145 | 15.5 | 0.0039 | 37.35 |
| 5 | 10 | 3 | 174 | 16 | 0.0021 | 51.29 |
| 6 | 12 | 4 | 258 | 17.5 | 14.285 | TL |

TL (tempo limite) = 3600s

nais, cujos resultados são apresentados na tabela 3.10, foram usadas duas instâncias com cinco itens, uma das quais com quatro itens ortogonais e um item retangular e outra com três itens ortogonais e dois retangulares. Foi também usada uma instância com seis itens, dos quais quatro são ortogonais e dois são retangulares. Depois da divisão dos itens ortogonais em retângulos, através do processo apresentado na figura 3.4, resultaram para a primeira instância nove retângulos, para a segunda dez retângulos e para a terceira doze retângulos. Os resultados apresentados na tabela 3.10 permitem mostrar que a existência itens ortogonais tem impacto na complexidade do modelo. As duas primeiras instâncias com itens ortogonais, apesar de apresentarem o mesmo número total de itens, o número de variáveis binárias da primeira é inferior ao da segunda. Esta situação é consequência do menor número de retângulos depois da divisão dos itens ortogonais. O tempo de resolução da primeira é inferior ao da segunda porque, para além de ter menos variáveis binárias, as posições de alguns retângulos estão fixas, por serem partes de itens ortogonais. Na instância com seis itens não foi obtida a solução ótima porque foi atingido o tempo limite (3600s). Apesar desta instância ter o mesmo número de itens ortogonais que a primeira, tem mais retângulos antes e depois da divisão dos itens ortogonais, resultando num aumento significativo do número de variáveis binárias. Na figura 3.11 apresenta-se o padrão obtido para a instância com quatro itens ortogonais e um retângulo.

O último conjunto de experiências teve como alvo o estudo do PPMO cujo modelo foi apresentado na secção 3.5. Recorde-se que os objetivos considerados são a minimização da área e a minimização da soma das distâncias

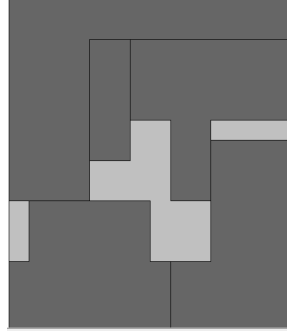


Figura 3.11: Exemplo de um padrão com 4 itens ortogonais e 1 retângulo

e pretende-se verificar se eles são ou não antagônicos. Cada um dos objetivos foi normalizado porque apresentam diferentes magnitudes. Usou-se para a sua normalização a expressão apresentada na equação (3.63).

$$\frac{Obj - Min_Obj}{Max_Obj - Min_Obj} \quad (3.63)$$

Na equação (3.63) Obj é o valor do objetivo que se pretende normalizar (área ou distâncias), o valor Min_Obj é determinado, resolvendo o modelo, considerando apenas um objetivo (o objetivo a normalizar) e o valor Max_Obj determina-se, resolvendo o modelo, com o outro objetivo considerado.

Os resultados destas experiências estão apresentados na tabela 3.11. Foram consideradas quatro instâncias, duas com sete itens retangulares com cinco e dez conjuntos de itens ligados entre si (também denominados de redes); e duas com oito itens, também retangulares, com dez e vinte conjuntos de ligações. Cada conjunto de ligações, para as instâncias com sete itens, tem no máximo três itens ligados enquanto que cada conjunto de ligações para as instâncias com oito itens tem no máximo cinco itens ligados. Foram escolhidas instâncias com sete e oito itens retangulares porque, tal como comprovam os resultados da primeira experiência, com este número de itens é possível resolver facilmente os modelos. Nestas tabelas apresentam-se o número total de itens e o número de conjuntos de ligações para cada instância. Para cada instância, são resolvidos vários modelos que diferem entre si na variação do valor de α (equação (3.62)). Nesta tabela apresentam-se também o tempo de execução de cada modelo e os respectivos valores do perímetro e da distância.

Na figura 3.12 apresenta-se os gráficos com as fronteiras eficientes para

Tabela 3.11: Fronteira eficiente do problema integrado de minimização da área e das distâncias

| # Ret. | # Redes | α | Tempo (s) | Perímetro | Distância |
|--------|---------|----------|-----------|-----------|-----------|
| 7 | 5 | 0.0 | 2.01 | 742 | 1277.5 |
| | | 0.1 | 4.85 | 742 | 1277.5 |
| | | 0.2 | 6.48 | 742 | 1277.5 |
| | | 0.3 | 5.51 | 742 | 1277.5 |
| | | 0.4 | 6.60 | 742 | 1277.5 |
| | | 0.5 | 6.33 | 742 | 1277.5 |
| | | 0.6 | 6.54 | 742 | 1277.5 |
| | | 0.7 | 6.00 | 749 | 1218.0 |
| | | 0.8 | 6.77 | 749 | 1218.0 |
| | | 0.9 | 7.03 | 784 | 1137.5 |
| | | 1.0 | 7.02 | 866 | 1099.0 |
| 7 | 10 | 0.0 | 1.87 | 742 | 2422.0 |
| | | 0.1 | 8.95 | 742 | 2422.0 |
| | | 0.2 | 8.41 | 742 | 2422.0 |
| | | 0.3 | 8.14 | 742 | 2422.0 |
| | | 0.4 | 7.69 | 742 | 2422.0 |
| | | 0.5 | 8.44 | 742 | 2422.0 |
| | | 0.6 | 10.65 | 742 | 2422.0 |
| | | 0.7 | 10.22 | 742 | 2422.0 |
| | | 0.8 | 11.43 | 749 | 2320.5 |
| | | 0.9 | 16.93 | 763 | 2170.0 |
| | | 1.0 | 17.13 | 789 | 2121.0 |
| 8 | 10 | 0.0 | 7.73 | 826 | 2446.5 |
| | | 0.1 | 59.93 | 826 | 2446.5 |
| | | 0.2 | 80.83 | 826 | 2446.5 |
| | | 0.3 | 61.91 | 826 | 2446.5 |
| | | 0.4 | 63.90 | 826 | 2446.5 |
| | | 0.5 | 109.09 | 826 | 2446.5 |
| | | 0.6 | 98.59 | 826 | 2446.5 |
| | | 0.7 | 147.72 | 826 | 2446.5 |
| | | 0.8 | 202.98 | 826 | 2446.5 |
| | | 0.9 | 191.22 | 840 | 2380.0 |
| | | 1.0 | 119.31 | 939 | 2282.0 |
| 8 | 20 | 0.0 | 3.21 | 826 | 5799.5 |
| | | 0.1 | 68.95 | 826 | 5799.5 |
| | | 0.2 | 26.70 | 826 | 5799.5 |
| | | 0.3 | 38.99 | 826 | 5799.5 |
| | | 0.4 | 122.53 | 826 | 5799.5 |
| | | 0.5 | 67.05 | 826 | 5799.5 |
| | | 0.6 | 87.97 | 826 | 5799.5 |
| | | 0.7 | 78.39 | 826 | 5799.5 |
| | | 0.8 | 73.79 | 847 | 5446.0 |
| | | 0.9 | 144.76 | 847 | 5446.0 |
| | | 1.0 | 131.18 | 973 | 5225.5 |

as quatro instâncias apresentadas na tabela 3.11. Os pontos representados nesses gráficos foram obtidos, tal como já referido, pelo modelo matemático apresentado na secção 3.5. Os gráficos desta figura¹ sugerem que o problema tem realmente uma componente multiobjetivo. Note-se que alguns pontos estão sobrepostos. A componente multiobjetivo é menor com o aumento do número de conjuntos de ligações entre os itens. É de salientar que nas instâncias com oito itens existem mais itens por conjunto de ligações, consequentemente uma pequena mudança no ponto de posicionamento de um dos itens pode ter um pequeno impacto no valor da função objetivo. Isto é justificado pelo uso do semiperímetro do retângulo exterior envolvente para medir a distância de cada conjunto de ligações. O mesmo não ocorre quando, por exemplo, existem apenas dois itens ligados, onde uma pequena mudança no posicionamento de um dos itens pode ter um grande impacto no valor da função objetivo.

Destes conjuntos de experiências computacionais são de realçar várias conclusões: o modelo global para a minimização da área do objeto não é resolúvel, em tempo útil, para instâncias de média e grande dimensão (mais de dez itens); este modelo, que lida apenas com itens retangulares, pode ser facilmente estendido para lidar com itens ortogonais. Foi também constatado que o tempo de execução do modelo com itens ortogonais é superior ao do modelo apenas com itens retangulares quando o número total de itens é o mesmo nas duas situações. Após a divisão dos itens ortogonais em retângulos e quando o número total de retângulos é o mesmo, a complexidade do modelo para itens ortogonais é inferior à do modelo para itens retangulares. Este resultado é consequência da existência de retângulos que têm posições fixas relativamente a outros. Foi também possível concluir que, quando se consideram simultaneamente os objetivos de minimizar a área do objeto e de minimizar a soma das distâncias, existe uma pequena componente multiobjetivo. As experiências realizadas sugeriram que o número de conjuntos de itens ligados e o número de itens ligados em cada conjunto tem alguma influência na componente MO.

¹Nestes gráficos não está representada a origem do referencial para facilitar a visualização dos valores representados.

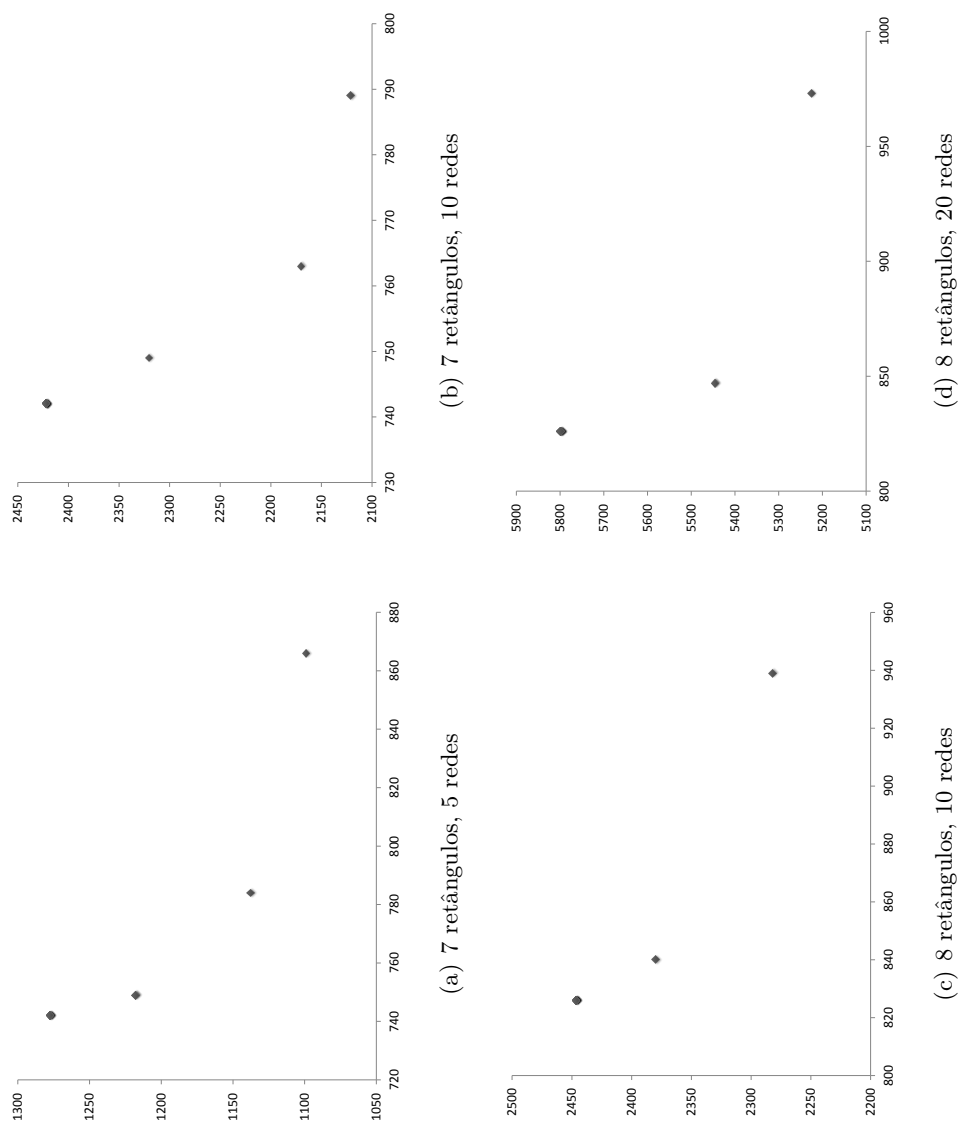


Figura 3.12: Fronteira eficiente para a minimização da área e da soma das distâncias

Capítulo 4

Heurísticas Construtivas para Problemas de Posicionamento de Minimização da Área

Em teoria, os problemas de otimização combinatória podem ser resolvidos enumerando as soluções da região admissível e calculando para cada solução o valor da função objetivo, obtendo-se sempre a solução ótima para o problema em questão. No entanto, o número de soluções da região admissível pode crescer de forma exponencial apenas com um pequeno aumento na dimensão do problema. Com efeito, esta constatação torna impossível resolver muitos problemas reais até à otimalidade, apesar da rápida evolução da tecnologia dos computadores atuais. A necessidade prática de resolver problemas de grande dimensão leva ao uso de alternativas aos algoritmos exatos, algoritmos que, apesar de não garantirem a solução ótima, obtêm boas soluções em tempos de cálculo aceitáveis. São exemplos destes algoritmos as heurísticas, os algoritmos de pesquisa local, as meta-heurísticas.

Neste capítulo são propostas duas heurísticas construtivas para PP de minimização da área. Na primeira secção apresenta-se uma heurística cujas soluções são padrões guilhotináveis (FAST-GR) que lida apenas com itens retangulares e, na segunda secção, é proposta uma heurística cujas soluções resultam em padrões não guilhotináveis (Rect-TOPOS) e onde os itens podem ter qualquer forma ortogonal.

4.1 Heurística de Construção de Padrões Guilhotináveis

Como abordagem ao PP de itens retangulares propomos a aplicação de uma heurística construtiva parametrizável, FAST-GR, com o objetivo de obter uma solução, de boa qualidade, num baixo tempo computacional. A ideia principal da heurística FAST-GR é unir sucessivamente pares de retângulos de modo a obter um retângulo exterior envolvente, que contém todos os itens, de área mínima. Esta heurística é parametrizável tendo também como objetivo adicional estudar se existe alguma combinação ou combinações de parâmetros que dominam ou se existem combinações que são sempre dominadas. Por exemplo, em alguns trabalhos afirma-se que os itens maiores devem ser posicionados em primeiro lugar, uma vez que estes itens podem, em muitas situações, determinar completamente as dimensões do padrão. Esta afirmação é feita baseada na intuição, não sendo apresentados quaisquer testes para a sua confirmação. Para além deste estudo, pretende-se simultaneamente verificar se os resultados dependem ou não do número total de retângulos e das relações entre as dimensões dos retângulos. Os testes computacionais que foram realizados com instâncias obtidas através de um gerador diferem entre si no número total e nas dimensões dos retângulos.

Uma versão preliminar deste trabalho encontra-se publicada no livro de atas do XLIII Simpósio Brasileiro de Pesquisa Operacional ([Oliveira et al., 2011b]).

4.1.1 Representação de Padrões Guilhotináveis

Para a representação de padrões guilhotináveis é usada uma estrutura em árvore e uma codificação desta estrutura semelhante à *Normalized Polish Expression*.

Nesta abordagem representamos um padrão por uma estrutura em árvore como se mostra na figura 4.1. A área sombreada na figura 4.1(c) representa o desperdício. Os números na árvore, figura 4.1(b), identificam os retângulos e os C'_i s representam o modo de unir dois retângulos. As folhas da árvore representam os retângulos iniciais (1,...,5) e cada união dá origem a um novo retângulo (6, 7 e 8).

Para a união de dois retângulos consideramos todas as formas de união possíveis, figura 4.2. Podemos observar, na figura 4.3, que para esta união,

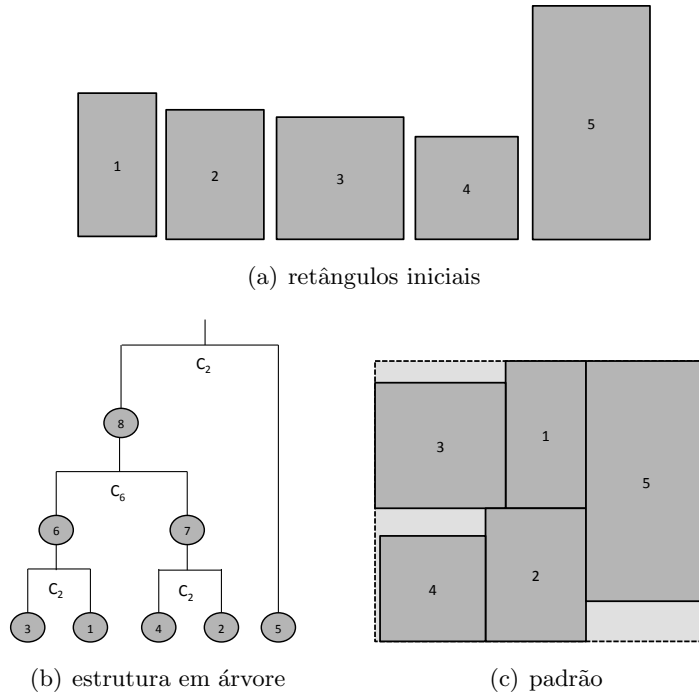


Figura 4.1: Heurística Construtiva FAST-GR

em padrões guilhotináveis, não interessa a posição relativa desses retângulos, i.e., não interessa qual dos retângulos está à esquerda ou à direita (acima ou abaixo) uma vez que o retângulo exterior envolvente é o mesmo.

Resolver o problema de posicionamento de retângulos é equivalente a determinar a estrutura em árvore correspondente ao padrão com valor mínimo da função objetivo. Para representarmos a estrutura em árvore usamos uma representação, semelhante à *Normalized Polish Expression*, onde é usado um par de sequências em vez de apenas uma sequência. Uma das sequências indica a ordem de escolha dos retângulos e a outra as operações usadas. Na abordagem proposta, a estrutura em árvore da figura 4.1(b) é representada pela seguinte codificação (3 1 4 2 6 7 8 5; 2 2 6 2) onde a primeira sequência indica a ordem pela qual os retângulos foram escolhidos e a segunda qual a operação usada para unir os retângulos. Por exemplo, os retângulos 3 e 1 são unidos segundo a operação 2. Na primeira sequência são também incluídos, para além dos retângulos iniciais, os retângulos resultantes de uniões. Diferentes sequências dão origem a diferentes árvores e, a partir da sequência de itens, a construção do padrão resulta de uma sequência de itens que é

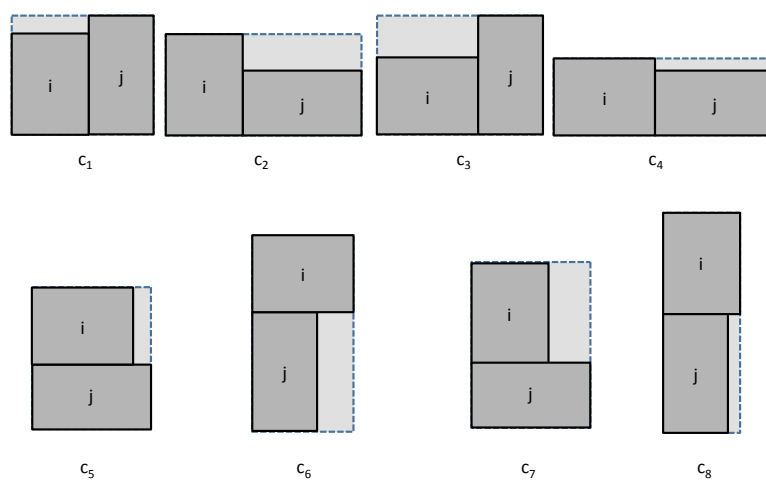


Figura 4.2: Operações usadas na união de dois retângulos

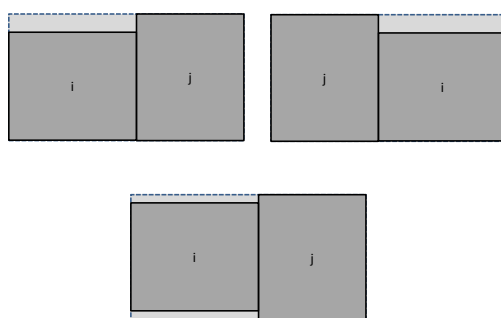


Figura 4.3: Posição relativa de dois retângulos no interior do retângulo exterior envolvente

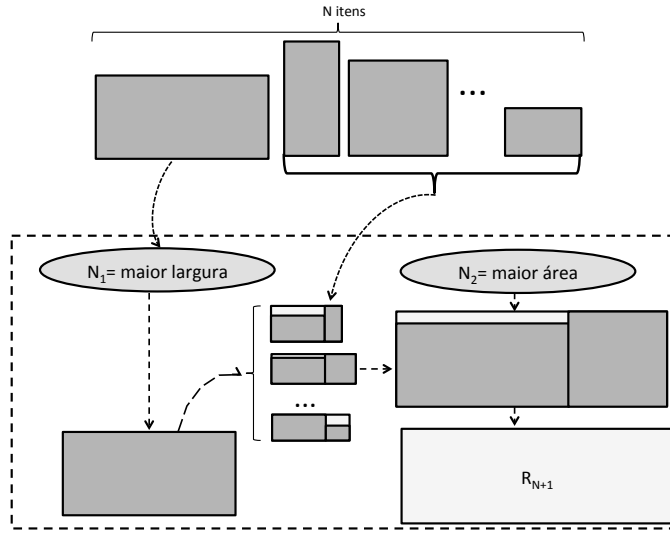


Figura 4.4: Seleção dos itens usando o parâmetro N_1 e o parâmetro N_2

obtida por uma estrutura em árvore.

4.1.2 Heurística FAST-GR

A ideia principal desta heurística é unir sucessivamente pares de retângulos de modo a que a área do retângulo exterior envolvente que contém todos os retângulos seja mínima. A abordagem proposta tem dois parâmetros de configuração que é necessário escolher, N_1 e N_2 . O parâmetro N_1 é usado para selecionar o primeiro dos retângulos a unir. O parâmetro N_2 é usado para selecionar o segundo retângulo cuja união com o primeiro resulta num retângulo que satisfaça o critério selecionado, figura 4.4.

O algoritmo inicia com um retângulo da lista de retângulos, que satisfaça N_1 . De seguida vai unir este retângulo com todos os outros da lista, ainda não usados, e escolhe aquele cujo retângulo resultante da união satisfaça o segundo parâmetro escolhido, N_2 . A união de dois retângulos é feita segundo uma das operações apresentadas na figura 4.2. O retângulo resultante da união na iteração k , R_{N+k} , é inserido na última posição da lista. O algoritmo termina quando na lista houver apenas um único retângulo ainda não usado. Este retângulo é o retângulo exterior envolvente. Na figura 4.5 apresenta-se um fluxograma com a descrição da heurística construtiva.

Para os parâmetros N_1 e N_2 foram usados critérios baseados na escolha

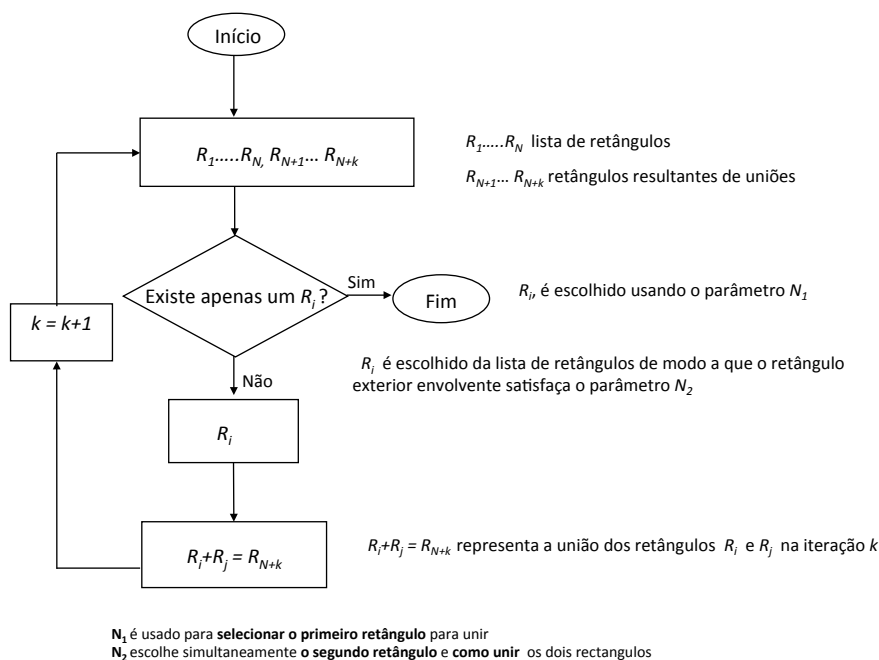


Figura 4.5: Fluxograma da heurística construtiva FAST-GR

do retângulo com maior ou menor largura, altura, área ou perímetro. Para o parâmetro N_2 foram ainda considerados o maior e o menor desperdício. A tabela 4.1 resume as várias possibilidades para os valores de N_1 e N_2 .

4.1.3 Testes Computacionais

Nesta secção apresentam-se os testes computacionais realizados para analisar o comportamento da heurística e estudar o efeito dos parâmetros de configuração nos resultados obtidos. Simultaneamente, pretende-se também

Tabela 4.1: Possibilidades de escolha para valores de N_1 e N_2 .

| | N_1 | N_2 |
|---------------|-------|-------|
| maior largura | + | + |
| menor largura | + | + |
| maior altura | + | + |
| menor altura | + | + |
| maior área | + | + |

| | N_1 | N_2 |
|-------------------|-------|-------|
| menor área | + | + |
| maior perímetro | + | + |
| menor perímetro | + | + |
| maior desperdício | | + |
| menor desperdício | | + |

+ é possível escolher o critério

verificar se os resultados dependem ou não das relações entre as dimensões dos retângulos (i.e. a forma dos retângulos) e se dependem do número total de retângulos considerados. Para esta análise foram usadas várias instâncias cuja construção se descreve a seguir. Para analisar a dependência das relações entre as dimensões dos retângulos foram geradas instâncias aleatórias com base em sete classes diferentes de instâncias. As primeiras quatro classes foram propostas por Martello e Vigo [1998] e são constituídas por retângulos de quatro tipos. Cada tipo consiste na geração de retângulos com largura e altura (w e h) dados por :

- Tipo 1 - w e h são uniformemente gerados em $[\frac{2}{3}W, W]$ e $[1, \frac{1}{2}H]$
- Tipo 2 - w e h são uniformemente gerados em $[1, \frac{1}{2}W]$ e $[\frac{2}{3}H, H]$
- Tipo 3 - w e h são uniformemente gerados em $[\frac{1}{2}W, W]$ e $[\frac{1}{2}H, H]$
- Tipo 4 - w e h são uniformemente gerados em $[1, \frac{1}{2}W]$ e $[1, \frac{1}{2}H]$

Considerou-se $W = H = 100$, com w e h números inteiros. Os retângulos do tipo 1 têm forma sobre o comprido, os do tipo 2 forma sobre o alto e os do tipo 3 e 4 forma sobre o quadrado grande e pequeno, respetivamente. As quatro primeiras classes são constituídas por retângulos dos quatro tipos anteriores em diferentes percentagens:

- Classe i - retângulos do tipo i gerados com probabilidade de 70% e os restantes com probabilidade 10% de cada um dos outros tipos;

As restantes três classes foram propostas por Berkey e Wang [1987], onde as dimensões dos retângulos (w e h) são números inteiros dados por:

- Classe 5 - w e h são uniformemente gerados em $[1, 10]$;
- Classe 6 - w e h são uniformemente gerados em $[1, 35]$;
- Classe 7 - w e h são uniformemente gerados em $[1, 100]$;

A principal diferença entre as classes propostas por Martello e Vigo [1998] e Berkey e Wang [1987] é a diversidade de formas dos itens. Dentro destas classes a principal distinção está na variedade de tamanhos, consequência da obrigatoriedade das dimensões serem números inteiros. Foram realizados testes para estudar se o número total de retângulos de cada instância tem

influência nos resultados obtidos e se há interação entre os parâmetros escolhidos. Para cada classe usaram-se instâncias com $N = 20, 50, 100, 250, 500$ e 1000 retângulos. Para cada uma das sete classes e para cada valor de N foram geradas 50 instâncias aleatórias. Para um dado número total de retângulos, após conjugação dos restantes fatores envolvidos, dispõe-se de 168 000 soluções no total, resultantes de $6N \times 50$ instâncias $\times 8N_1 \times 10N_2 \times 7$ classes. Os resultados obtidos deveriam ser comparados com a solução ótima de cada instância, no entanto, como não é possível obtê-la para todas as instâncias, decidimos comparar com 100% da área usada, ou seja desperdício nulo. A heurística foi implementada na linguagem C++ e os testes decorreram num computador Intel Core i5 2.27 GHz com 4 GB de memória RAM.

De seguida, apresentam-se os resultados dos testes realizados que se basearam em comportamentos médios. Para analisar a influência dos parâmetros N_1 e N_2 no valor da função objetivo fez-se uma representação segundo diagramas de tipo caixa para se verificar se há opções que são claramente dominadas. Para esta verificação usou-se uma regra que se considerou ser forte, que é a comparação da mediana com o primeiro quartil, isto é, uma opção domina a outra se a sua mediana for melhor do que o 1º quartil da outra. Simultaneamente pretende-se também verificar se o número total de retângulos e as classes usadas influenciam os resultados fazendo-se para tal uma análise análoga à anterior.

Nas figuras 4.6–4.9 faz-se uma representação do valor da função objetivo para cada uma das classes consideradas, para o número total de retângulos e para os parâmetros N_1 e N_2 .

Da representação apresentada na figura 4.6 não é possível concluir que alguma classe domina as outras ou que alguma é dominada pelas outras classes.

Da figura 4.7 podemos igualmente concluir que o número total de retângulos (N) em cada instância não tem influência nos resultados.

Para o parâmetro N_1 , figura 4.8, não é óbvio, por simples visualização dos diagramas de tipo caixa, que existam comportamentos diferenciados. Já o gráfico da figura 4.9 sugere comportamentos diferenciados no parâmetro N_2 , em que os piores valores correspondem à escolha do retângulo com maior área, com maior perímetro e com maior desperdício. Para este parâmetro verifica-se também a existência de superioridade na escolha do retângulo exterior envolvente com menor área, com menor perímetro e com menor

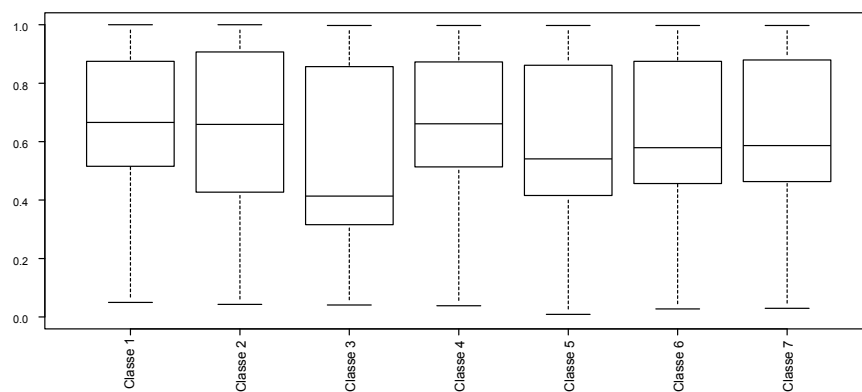


Figura 4.6: Variação do valor da função objetivo para cada classe considerada

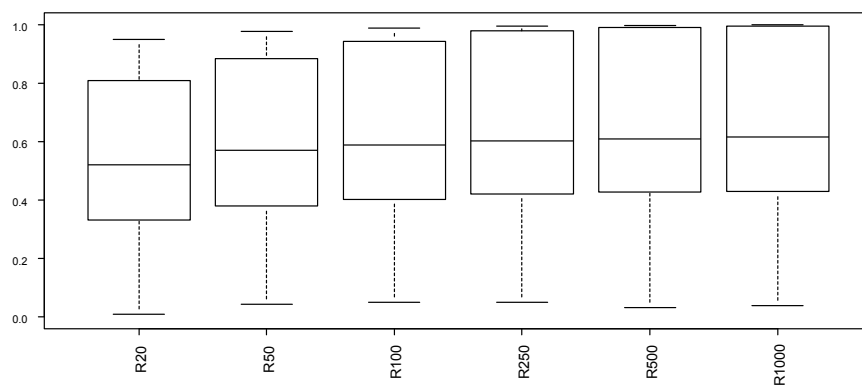


Figura 4.7: Variação do valor da função objetivo com o número total de retângulos

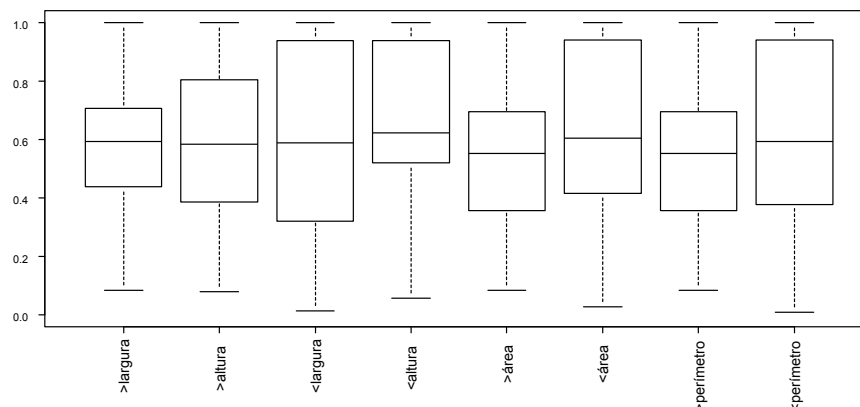


Figura 4.8: Variação do valor da função objetivo com o parâmetro N_1

desperdício, pois a mediana destas opções é melhor do que o primeiro quartil das restantes, levando apenas a considerar N_2 igual a estas escolhas nos testes que se seguem.

Para o parâmetro N_1 , constatou-se anteriormente que não era óbvio que existissem comportamentos diferenciados considerando todas as opções do parâmetro N_2 , figura 4.8. Interessa agora distinguir concretamente, no caso em que para N_2 se consideram apenas as opções de menor área, menor perímetro e menor desperdício se há alguma opção de N_1 que se revele melhor do que as restantes, figura 4.10. Esta figura mostra a existência de comportamentos diferenciados no valor da função objetivo em que os piores valores correspondem à escolha do retângulo com menor altura para o parâmetro N_1 , uma vez que o primeiro quartil desta escolha é pior do que a mediana de todas as outras opções. Logo, nos testes seguintes, não se irá considerar esta escolha para este parâmetro.

Para além de se avaliar a influência dos parâmetros da heurística nos valores da função objetivo pretendia-se também verificar a influência das classes nos resultados. Nas figuras 4.11–4.14 apresentam-se diagramas de tipo caixa para cada uma das sete classes, considerando para o parâmetro N_1 todas as opções à exceção da menor altura e para N_2 as melhores opções anteriormente referidas (menor área, menor perímetro e menor desperdício). A reta horizontal representada nestes gráficos tem valor na função objetivo igual à melhor mediana encontrada para as opções representadas nesse grá-

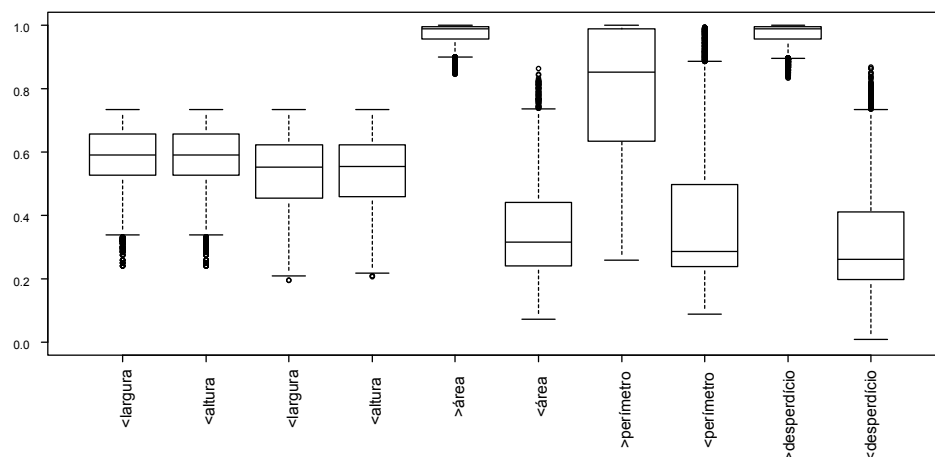


Figura 4.9: Variação do valor da função objetivo com o parâmetro N_2

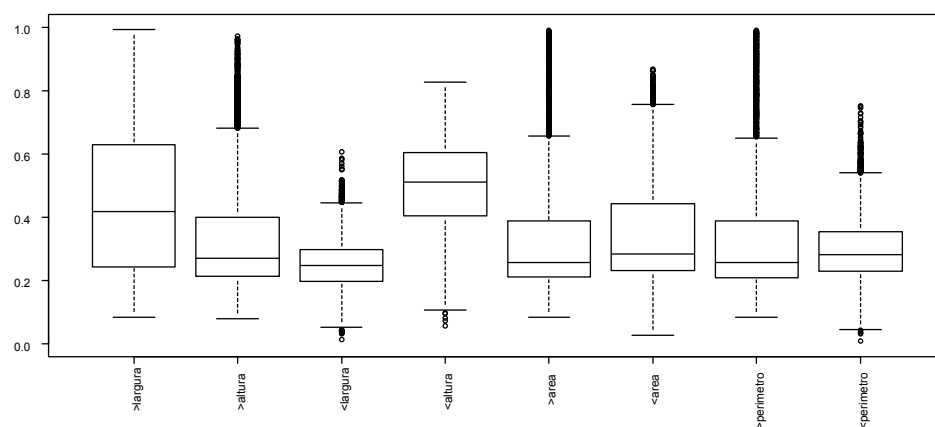


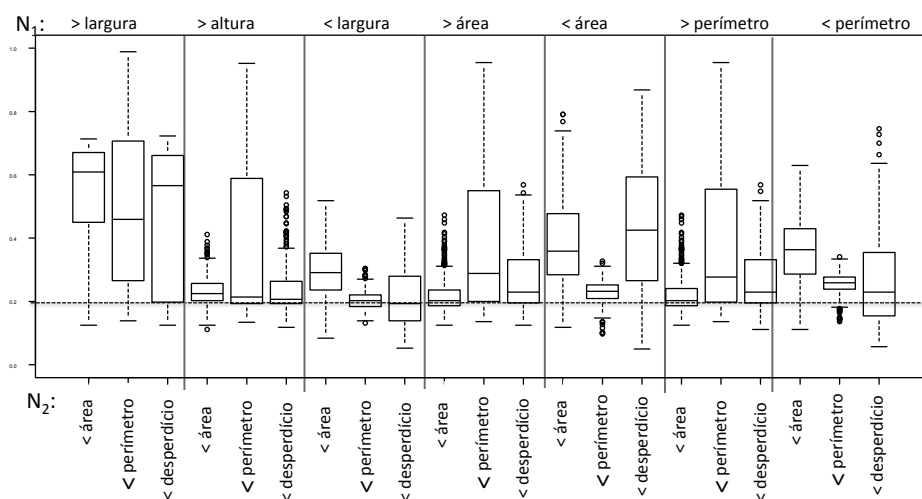
Figura 4.10: Variação do valor da função objetivo para N_1 considerando N_2 igual a <área, <perímetro e <desperdício

fico. Esta reta permite uma fácil comparação do 1º quartil das restantes opções com a mediana. Elimina-se uma opção para o parâmetro N_1 quando todas as opções de N_2 são dominadas pela melhor combinação. Através de uma análise dos diagramas representados nas figuras 4.11–4.14 pode ser eliminada a opção de escolha de menor área para o parâmetro N_1 para todas as classes. Para além desta opção podem eliminar-se para a classe 1 a maior largura e a maior altura (diagrama da figura 4.11(a)); maior largura para a classe 2 (diagrama da figura 4.11(b)) e as opções maior largura, maior altura, maior área e maior perímetro para as classes 4, 5, 6 e 7 (diagramas das figuras 4.12(b), 4.13 e 4.14). As melhores opções são: para a classe 1 menor largura com menor desperdício (diagrama da figura 4.11(a)); para a classe 2 maior altura com menor desperdício (diagrama da figura 4.11(b)); para a classe 3 maior altura com menor desperdício, maior área com menor área e menor desperdício e maior perímetro com menor área e menor desperdício (diagrama da figura 4.12(a)); e para as classes 4, 5, 6 e 7 menor largura com menor desperdício (diagramas das figuras 4.12(b), 4.13 e 4.14), para os parâmetros N_1 e N_2 , respetivamente.

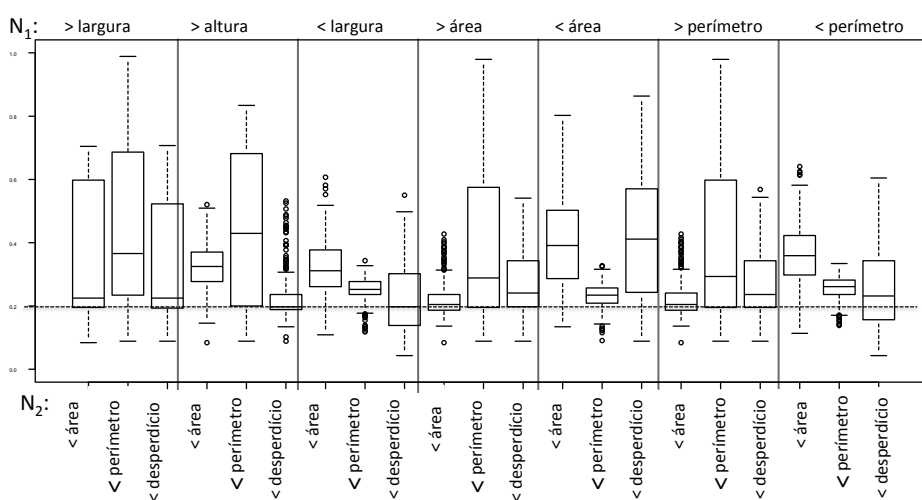
Para além de avaliarmos a influência dos parâmetros da heurística nos valores da função objetivo pretendia-se também verificar a influência do número de retângulos nos resultados. Para avaliar a influência do número de retângulos nos resultados fez-se uma análise análoga às anteriores. Desagregaram-se os valores por classe de instância para analisar se este número tem influência nas classes usadas e fez-se uma representação do número total de retângulos segundo diagramas de tipo caixa. Estes diagramas apresentam-se nas figuras 4.15 e 4.16. Destas figuras podemos concluir que o número total de retângulos é independente da classe de instância usada.

4.1.4 Comentários e Conclusões

Neste trabalho desenvolveu-se uma heurística construtiva para o problema de posicionamento de retângulos com a restrição do padrão obtido ser guilhotinável. A heurística construtiva proposta tem dois parâmetros de configuração que é necessário escolher, N_1 e N_2 . Foi realizada uma análise estatística com o objetivo de estudar o efeito destes parâmetros nos resultados obtidos. Estes parâmetros servem para escolher o par de retângulos a unir e a forma de os unir. N_1 é usado para selecionar o primeiro retângulo e N_2 é usado para selecionar o segundo retângulo cuja união com o

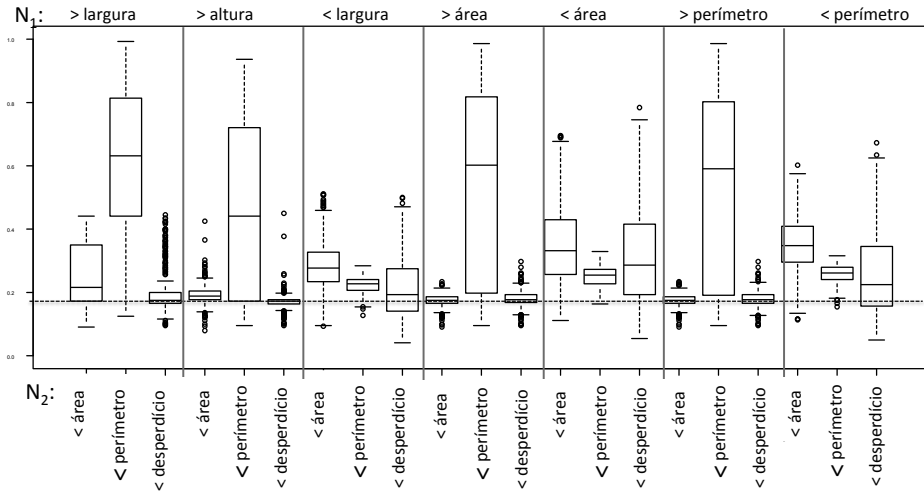


(a)

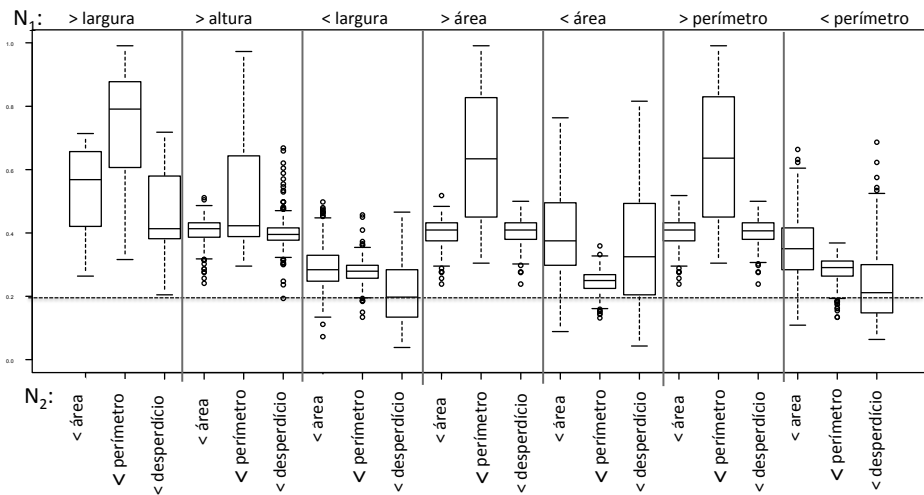


(b)

Figura 4.11: Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continua)

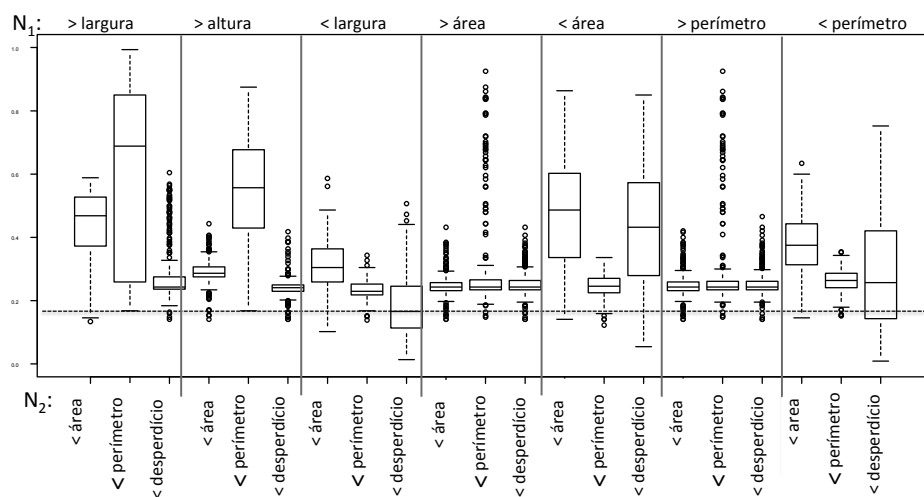


(a)

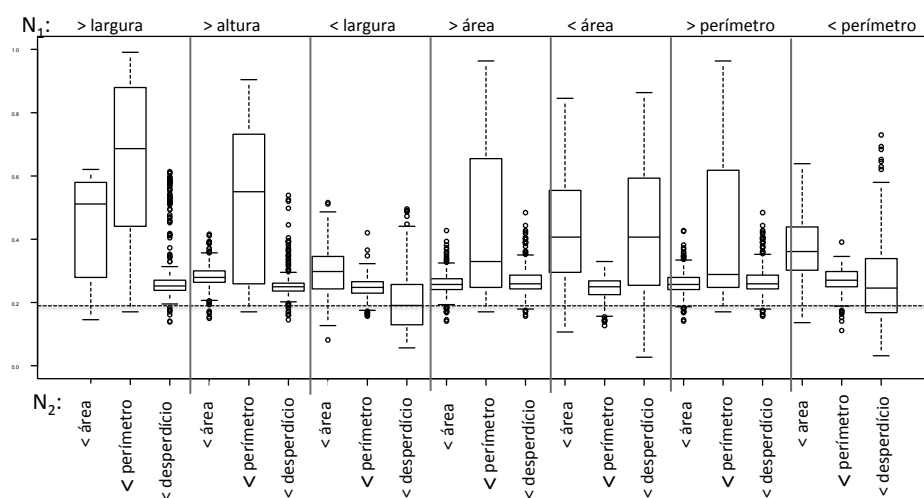


(b)

Figura 4.12: Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continuação e continua)



(a)



(b)

Figura 4.13: Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continuação e continua)

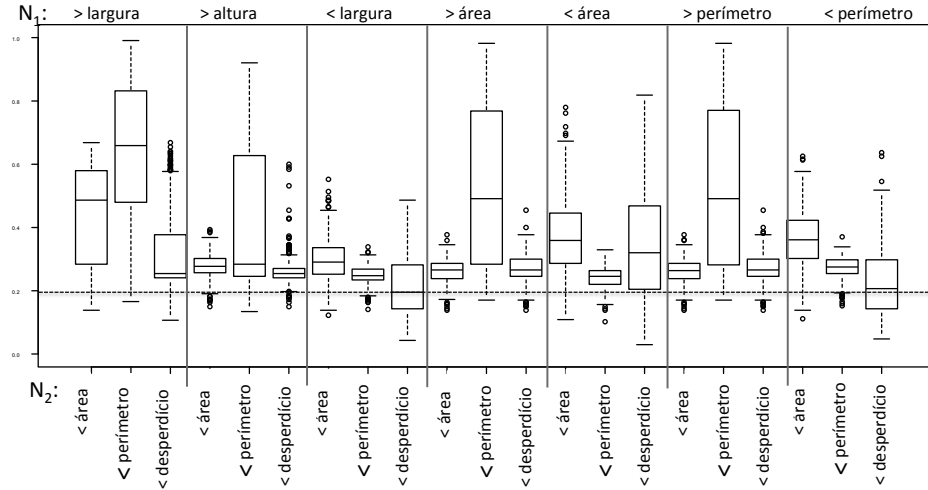


Figura 4.14: Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continuação)

primeiro resulte num retângulo que satisfaça o critério selecionado. Foi feita uma comparação entre os diferentes critérios o que nos permitiu identificar os melhores e os piores. Conseguimos limitar as opções para N_2 concluindo que as melhores opções são a obtenção do retângulo exterior envolvente com menor área, menor perímetro e menor desperdício, o que é bastante intuitivo. Para o parâmetro N_1 foi possível eliminar a escolha dos retângulos com menor área, para todas as classes e eliminar outras combinações que variam com as classes de instância. Foi também possível concluir que as melhores e as piores combinações para os parâmetros N_1 e N_2 são as mesmas para as classes 4, 5, 6 e 7. Tal como esperado, os resultados mostraram que, independentemente da classe usada, para o parâmetro N_2 , o critério menor desperdício está sempre entre as melhores opções. Na tabela 4.2 apresenta-se um resumo com as melhores e as piores combinações entre os parâmetros N_1 e N_2 .

A análise estatística serviu também para estudar a influência do número total de retângulos por instância e a influência da relação entre as dimensões dos retângulos no valor da função objetivo. Conseguimos obter uma heurística construtiva parametrizável com duas combinações que são independentes da classe de instância usada: menor largura para N_1 com menor

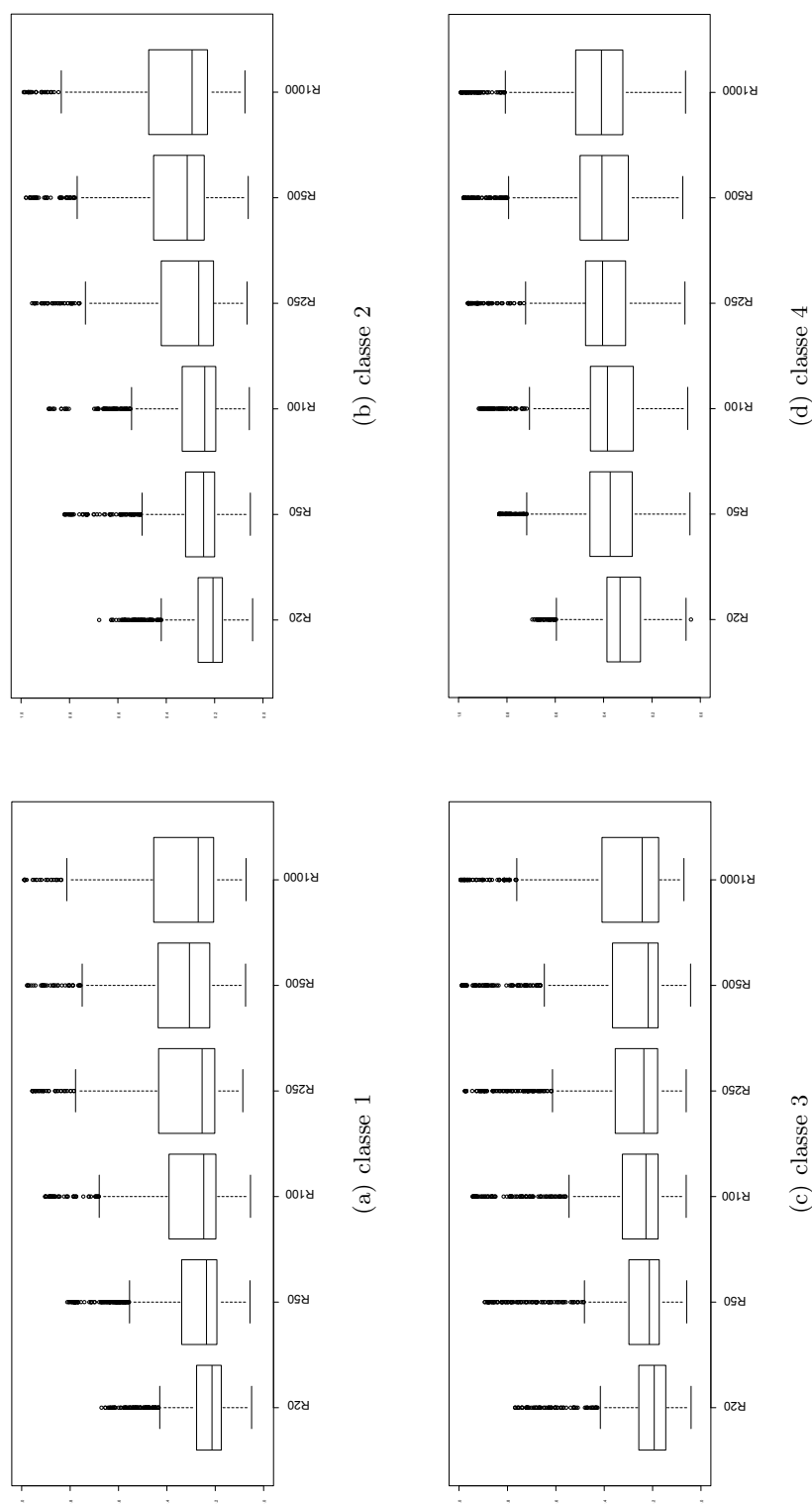


Figura 4.15: Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continua)

Figura 4.16: Variação do valor da função objetivo com N_1 e N_2 , número total de retângulos e classes (continuação)

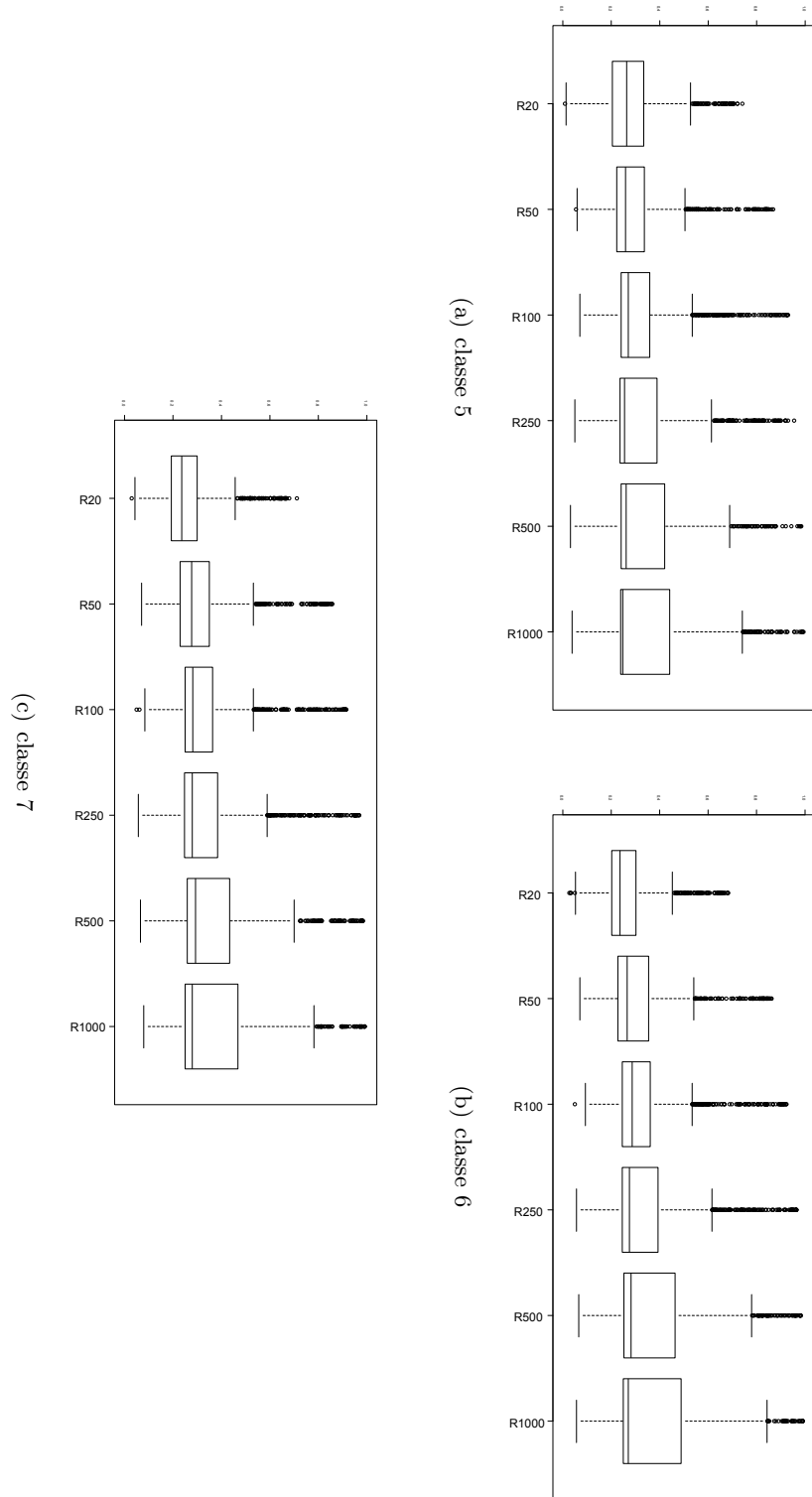


Tabela 4.2: Tabela resumo das combinações N_1 e N_2 .

| Classes | N_2 | N_1 | | | | | | | |
|------------|---------------|----------|----------|---------|-------|-------|------------|------------|---|
| | | >largura | <largura | >altura | >área | <área | >perímetro | <perímetro | |
| 1 | < área | x | | x | + | x | + | | |
| | < perímetro | x | + | x | | x | | | |
| | < desperdício | x | + | x | + | x | + | | + |
| 2 | < área | + | | | + | | + | | |
| | < perímetro | | | | + | x | + | | |
| | < desperdício | + | + | + | + | x | + | | + |
| 3 | < área | + | | + | + | x | + | | |
| | < perímetro | | | | | x | | | |
| | < desperdício | + | + | | + | x | + | | + |
| 4, 5, 6, 7 | < área | x | | x | x | x | x | | |
| | < perímetro | x | | x | x | x | x | | |
| | < desperdício | x | + | x | x | x | x | | + |

+ combinação pertencente ao conjunto das melhores combinações
X combinação pertencente ao conjunto das piores combinações

desperdício para N_2 e menor perímetro para N_1 com menor desperdício para N_2 . Como o tempo computacional para correr cada combinação, por instância, é inferior a um segundo, é possível correr consecutivamente as duas variantes e escolher a solução com o melhor resultado.

4.2 Heurística de Construção de Padrões Não Guihotináveis

Nesta secção apresenta-se uma heurística construtiva, para o posicionamento de itens ortogonais. A ideia principal desta heurística, Rect-TOPOS, consiste em adicionar sucessivamente um item à volta de uma solução parcial (conjunto de itens posicionados) com o objetivo de se obter o retângulo exterior envolvente de área mínima. Para o posicionamento deste novo item são usados diferentes critérios para a escolha do melhor ponto de posicionamento e da sua orientação. Esta heurística opera diretamente no padrão não sendo necessária nenhuma codificação adicional das soluções. Uma das grandes vantagens desta heurística é a possibilidade de preenchimento de espaços vazios no meio do padrão. Outra vantagem resulta da possibilidade dos itens serem posicionados à volta da solução parcial, sendo uma heurística bastante adequada ao problema que é alvo de estudo (PP de minimização da área do objeto).

Este trabalho encontra-se publicado, em Português, no livro de atas do 15º Congresso da Associação Portuguesa de Investigação Operacional – IO 2011 ([Oliveira et al., 2011d]) e, em Inglês, no livro de atas do VII ALIO/EURO – Workshop on Applied Combinatorial Optimization ([Oliveira et al., 2011c]).

4.2.1 Invólucro de Posicionamento

Em qualquer PP é necessário assegurar que não existem sobreposições entre itens, garantido a existência de soluções admissíveis. Quando se pretende posicionar um item, para garantir que este item não se sobrepõe a outro, é necessário determinar uma região admissível de posicionamento. Para determinar esta região foi proposto por Art [1966] o conceito do invólucro de posicionamento (*no-fit polygon*). O invólucro de posicionamento, NFP_{ij} , de um item j (item orbital) em relação a um item i , mantido fixo, é o lugar geométrico do percurso traçado por um ponto do item j (ponto de referên-

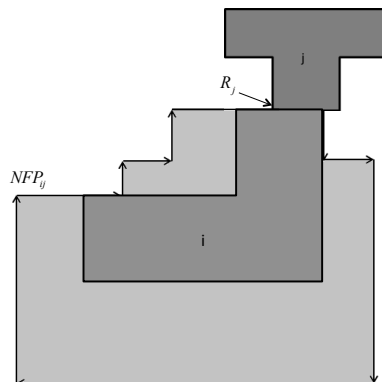


Figura 4.17: Invólucro de posicionamento do item j em relação ao item i (NFP_{ij})

cia), quando este se desloca à volta do item i . Durante este movimento, as orientações relativas entre os dois itens são mantidas, sem nunca perderem o contacto. Um exemplo da determinação do invólucro de posicionamento encontra-se representado na figura 4.17. Nesta figura, o invólucro de posicionamento NFP_{ij} é representado por um polígono orientado, cujo exterior corresponde ao lado esquerdo das arestas. Assim, existem três situações que podem surgir, considerando que o ponto de referência do item j (R_j) está em relação ao invólucro de posicionamento (NFP_{ij}):

- no interior - existe sobreposição entre os itens i e j ;
- na fronteira - resulta num contacto entre o item j e o item i ;
- no exterior - não ocorre nenhuma das situações anteriores, isto é, os itens i e j não se sobrepõem nem estão em contacto.

Com este conceito é possível simplificar o problema, que resulta em assegurar a não sobreposição entre pares de itens, pois consegue-se transformar comparações entre itens em comparações entre um ponto (ponto de referência) e um polígono (invólucro de posicionamento). É possível calcular o invólucro de posicionamento, entre todos os pares de itens, numa fase de pré-processamento.

4.2.2 Heurística Rect–TOPOS

Para a resolução do PP de minimização de área propomos uma heurística construtiva, Rect–TOPOS. Esta heurística é uma variante da heurística TOPOS que foi originalmente desenvolvida para resolver problemas com itens cujas formas são irregulares¹, Oliveira et al. [2000]. A ideia principal da heurística TOPOS é adicionar sucessivamente um novo item à volta de uma solução parcial, i.e., um conjunto de itens já posicionados (figura 4.18). A escolha do item a posicionar e do respetivo ponto de posicionamento é efetuada em dois níveis. Num primeiro nível são testados todos os itens a posicionar e todas as orientações admissíveis, num conjunto finito de pontos de posicionamento possíveis, e escolhido um ponto de posicionamento para cada par item/orientação. Esta escolha baseia-se num dos seguintes critérios:

- minimização da envolvente retangular da nova solução parcial;
- minimização do comprimento da nova solução parcial;
- maximização da sobreposição entre as envolventes retangulares do novo item e da solução parcial.

A escolha do par item/orientação efetivamente a posicionar é realizada num segundo nível com base nos critérios:

- diferença entre a área da envolvente retangular da solução parcial e a área de todos os itens já posicionados (desperdício);
- sobreposição entre a envolvente retangular do item em avaliação e a envolvente retangular de cada item já posicionado;
- distância euclidiana entre o centro da envolvente retangular do item em avaliação e o centro da envolvente retangular da solução parcial.

O objetivo é minimizar o comprimento do padrão já que nestes problemas a largura é fixa.

Na variante Rect–TOPOS usamos a mesma ideia geral da heurística TOPOS, selecionando e acrescentando, sucessivamente, um novo item que pode ser ortogonal ou retangular, à solução parcial, com o objetivo de obter o retângulo exterior envolvente de área mínima que contém todos os itens.

¹Itens com ângulos interiores arbitrários.

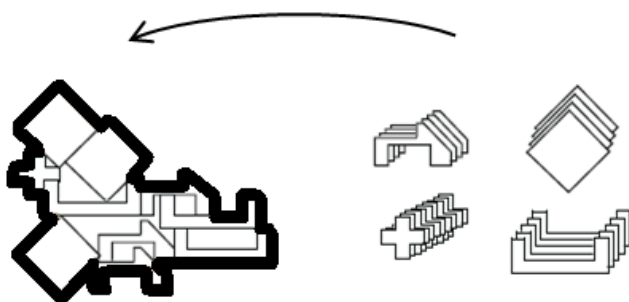


Figura 4.18: Construção da solução parcial Oliveira et al. [2000]

Optamos por utilizar um único nível para encontrar o próximo item a posicionar, a sua orientação e o ponto de posicionamento. A existência de um só nível permite que se escolha o melhor item para posicionar de entre todas as possibilidades, contrariamente ao que acontece quando há dois níveis em que existe uma seleção inicial do ponto de posicionamento para cada item a posicionar. Os critérios usados para efetuar esta escolha foram:

- minimização do *desperdício* da envolvente retangular resultante, figura 4.19(a);
- minimização da *distância* do centro da envolvente retangular do item a posicionar ao centro da envolvente retangular dos itens já posicionados, figura 4.19(b);
- minimização do *perímetro* da envolvente retangular resultante, figura 4.19(c).

Os dois primeiros critérios são retirados diretamente dos critérios utilizados no segundo nível do TOPOS, tendo sido introduzido um novo critério - a minimização do perímetro da envolvente retangular que contém o item em avaliação e a solução parcial, figura 4.19(c).

O terceiro critério utilizado no TOPOS (minimização da sobreposição entre a envolvente retangular do novo item a posicionar e a envolvente retangular de cada item já posicionado) não foi considerado uma vez que não é adequado a situações onde nos itens a posicionar existe um grande número de retângulos. Nestas situações, como a envolvente retangular de um retângulo é o próprio retângulo, não faz sentido tentar maximizar a sobreposição de dois retângulos pois não é permitida a sobreposição de itens (figura 4.20).

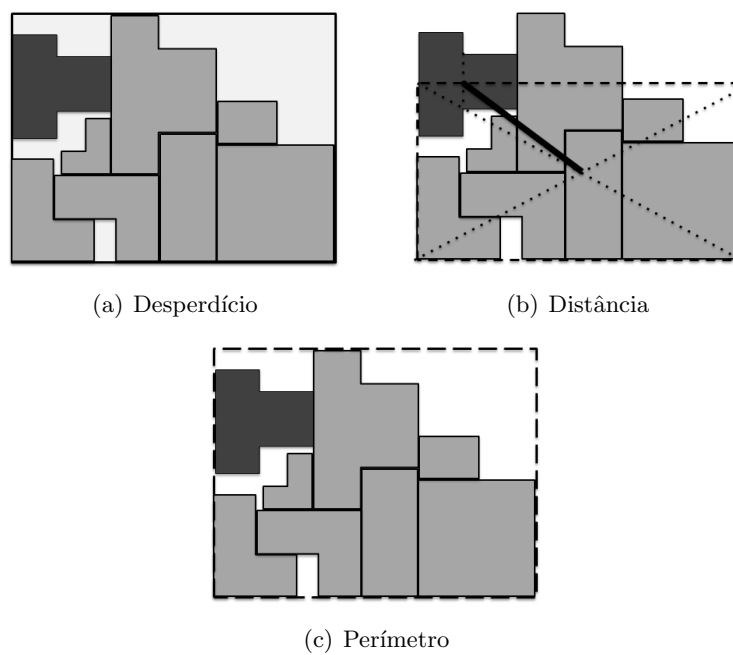


Figura 4.19: Critérios usados na escolha do próximo item a posicionar



Figura 4.20: Critério da sobreposição entre retângulos

Tal como acontece no TOPOS, o processo iterativo necessita de selecionar um primeiro item, para iniciar a solução parcial. Para esta seleção optamos por usar critérios que privilegiam a seleção de itens "grandes":

- item com maior *área*;
- item com maior *perímetro*;
- item com maior *largura*.

4.2.3 Testes Computacionais

Nesta secção serão apresentados os resultados computacionais obtidos com a heurística Rect-TOPOS. Os testes computacionais foram realizados numa estação de trabalho Linux equipada com duplo processador Intel XEON Dual Core 5160, 3GHz. Apesar de a estação de trabalho possuir dois CPUs, foi utilizado apenas um processador nos testes. Para avaliar a heurística Rect-TOPOS, foram utilizados quatro conjuntos de instâncias de teste, tendo as instâncias de um deles sido geradas artificialmente, e as dos três conjuntos restantes foram retiradas de publicações. Foram utilizados os seguintes quatro conjuntos de instâncias:

- instâncias do conjunto de referência MCNC² com origem no projeto de circuitos VLSI, no qual todos os itens têm forma retangular e onde o número total de itens a posicionar não ultrapassa os 50 (APTE, XEROX, HP, AMI33, AMI49);
- instâncias que contêm também apenas retângulos, no entanto, diferem das anteriores por apresentarem um número total de itens entre 100 e 500, (RP100, RP200, PCB146, PCB500)³;
- instâncias de Nakatake e Fujiyoshi [1997] (NAKATAKE1, NAKATAKE2), (LIN) e Xu et al. [1998](AMI49L, AMI49LT) que contêm um misto de itens que são retângulos, itens com forma em L e/ou T e polígonos ortogonais (U, +, H, em escada);
- instâncias geradas a partir das instâncias AMI33 e AMI49 do conjunto de referência MCNC (AMI33LTa, AMI49LTa). Nestas instâncias usou-se

²<http://vlsicad.eecs.umich.edu/BK/MCNCbench/HARD/>

³<http://www.simplex.tu-tokyo.ac.jp/imahori/packing/>

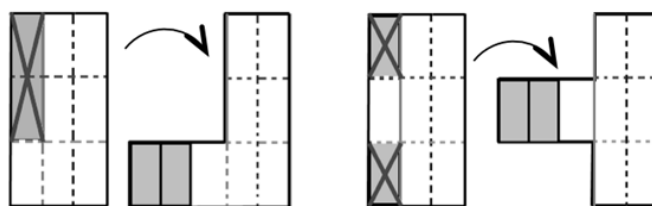


Figura 4.21: Construção de itens ortogonais nas instâncias geradas

como regra alterar aproximadamente 10% do número total de retângulos para itens com formas em L e/ou T. Cada item é construído de forma a ter dimensões inteiras e a sua área seja aproximada da área do retângulo que lhe deu origem, figura 4.21.

As instâncias de teste utilizadas apresentam características variadas, nomeadamente no número total de itens, no número de itens com formas diferentes (número de tipos de itens), na forma dos itens (retangular, em L, em T e outros polígonos ortogonais). Estas características encontram-se resumidas na Tabela 4.3.

A Tabela 4.4 resume os testes computacionais efetuados para avaliar a heurística Rect-TOPOS. Foram testados os três critérios para a escolha do próximo item a posicionar, a sua orientação e o ponto de posicionamento, anteriormente apresentados (desperdício, distância e perímetro) e, para cada um deles, foram consideradas três possibilidades, área, perímetro e largura, para a escolha do item que inicia a solução parcial. Os valores indicados na tabela são os aproveitamentos obtidos, medidos como o quociente entre a área dos itens posicionados e a área da envolvente retangular obtida. A negrito indica-se o melhor resultado obtido para cada instância. Nesta tabela apresenta-se também, para cada instância, o tempo médio de resolução (T_m). Os tempos de processamento não incluem o tempo de pré-processamento utilizado no cálculo do invólucro de posicionamento, entre pares de itens. Este cálculo é demorado pois foi desenvolvido para lidar com itens irregulares. Refira-se que os tempos de resolução, para cada uma das combinações dos critérios, aplicadas a cada instância, não apresentam grande variabilidade. Finalmente apresentam-se ainda, para cada instância, o melhor resultado encontrado na literatura: aproveitamento (Aprov.), tempo de resolução (T) e a indicação da referência à publicação.

Tabela 4.3: Características das instâncias usadas.

| Instância | # Itens | | | |
|-----------|---------|---------|---------|--------|
| | # Total | # Tipos | # Rect. | # Ort. |
| APTE | 9 | 3 | 9 | — |
| XEROX | 10 | 10 | 10 | — |
| HP | 11 | 6 | 11 | — |
| AMI33 | 33 | 31 | 33 | — |
| AMI49 | 49 | 46 | 49 | — |
| RP100 | 100 | 99 | 100 | — |
| PCB146 | 146 | 22 | 200 | — |
| RP200 | 200 | 99 | 146 | — |
| PCB500 | 500 | 417 | 500 | — |
| AMI49L | 28 | 28 | 7 | 21 |
| AMI49LT | 27 | 27 | 6 | 21 |
| NAKATAKE1 | 40 | 35 | 30 | 10 |
| NAKATAKE2 | 19 | 19 | 5 | 14 |
| LIN | 29 | 21 | 22 | 7 |
| AMI33LTa | 33 | 31 | 30 | 3 |
| AMI49LTa | 33 | 46 | 41 | 5 |

Tabela 4.4: Comparação dos resultados obtidos pela heurística Rect-TOPOS vs resultados da literatura.

| Instancia | Desperdício | | | Distância | | | Perímetro | | | Média Tempo (s) | Melhor resultado | |
|-----------|-------------|---------------|--------------|-------------|---------------|--------------|-------------|---------------|--------------|-----------------------|------------------|--------------|
| | Área (%) | Perim. (%) | Larg. (%) | Área (%) | Perim. (%) | Larg. (%) | Área (%) | Perim. (%) | Larg. (%) | | Aprov. (%). | Tempo (s) |
| APTE | 91.7 | 91.7 | 91.7 | 89.3 | 89.3 | 89.3 | 89.4 | 89.4 | 89.4 | 0.01 | 99.23 | 1.00 (a) |
| XEROX | 80.1 | 80.1 | 80.1 | 80.4 | 80.4 | 80.4 | 78.8 | 78.8 | 78.8 | 0.09 | 97.75 | 9812 (b) |
| HP | 84.8 | 84.8 | 69.5 | 83.4 | 83.4 | 69.5 | 92.4 | 92.4 | 93.6 | 0.03 | 98.70 | 891.0 (b) |
| AMI33 | 81.3 | 81.3 | 87.5 | 71.2 | 71.2 | 74.5 | 83.2 | 83.2 | 86.3 | 0.84 | 99.01 | 116.0 (a) |
| AMI49 | 80.7 | 80.7 | 80.7 | 79.2 | 79.2 | 79.2 | 84.3 | 84.3 | 84.3 | 1.97 | 98.58 | 1752.0 (a) |
| RP100 | 81.9 | 81.9 | 85.7 | 72.1 | 72.1 | 77.3 | 92.4 | 92.4 | 90.5 | 9.35 | 99.06 | 59.0 (a) |
| PCB146 | 62.2 | 62.2 | 62.2 | 78.6 | 78.6 | 78.6 | 88.1 | 88.1 | 88.1 | 0.95 | 98.85 | 2250.0 (a) |
| RP200 | 87.6 | 87.6 | 87.8 | 74.6 | 74.6 | 75.4 | 92.9 | 92.9 | 91.3 | 13.2 | 99.11 | 14.0 (a) |
| PCB500 | 86.5 | 86.5 | 86.5 | 78.1 | 78.1 | 78.1 | 96.7 | 96.7 | 96.7 | 221.0 | 99.07 | 554.0 (a) |
| AMI49L | 62.5 | 62.5 | 66.7 | 76.1 | 76.1 | 76.1 | 82.9 | 82.9 | 79.2 | 1.11 | 95.6 | 2728 (c) |
| AMI49LT | 73.1 | 73.1 | 66.3 | 78.7 | 78.7 | 75.3 | 79.3 | 79.3 | 82.3 | 1.08 | 95.1 | 2843 (c) |
| NAKATAKE1 | 82.5 | 82.5 | 76.3 | 80.7 | 80.7 | 78.4 | 85.2 | 85.2 | 86.7 | 1.35 | 96.9 | 10.24 (c) |
| NAKATAKE2 | 59.0 | 61.9 | 59.0 | 66.9 | 61.4 | 66.9 | 73.1 | 77.0 | 73.1 | 2.14 | 80.2 | 93.32 (c) |
| LIN | 78.9 | 74.0 | 78.0 | 78.0 | 78.0 | 74.0 | 84.0 | 84.0 | 91.0 | 0.82 | 100.0 | 2.72 (c) |
| AMI33LTa | 76.4 | 76.4 | 71.1 | 75.2 | 75.2 | 73.1 | 83.2 | 83.2 | 84.4 | 1.05 | — | — |
| AMI49LTa | 87.5 | 87.5 | 87.5 | 76.1 | 76.1 | 76.1 | 88.1 | 88.1 | 88.1 | 2.41 | — | — |

(a) Bortfeldt [2013]; (b) Chan e Markov [2004]; (c) Chen et al. [2010]

Da Tabela 4.4 podemos constatar que os melhores resultados foram obtidos quando se usa para a escolha do próximo item a posicionar e respetivo ponto de posicionamento o critério perímetro, à exceção das instâncias APTE e XEROX. Estas duas instâncias são muito sensíveis à escolha do primeiro item a posicionar, já que têm um reduzido número de itens a posicionar, nove e dez, respetivamente. Relativamente à escolha do primeiro item, os resultados mostram um maior equilíbrio entre os três critérios. Na comparação dos resultados obtidos com os melhores resultados publicados deverá ter-se em consideração que o Rect-TOPOS é apenas uma heurística construtiva, enquanto os melhores resultados publicados foram obtidos com abordagens baseadas em pesquisa local e em algoritmos de pesquisa em árvore. Em relação à comparação dos tempos de execução, a informação insuficiente, acerca das características do processador usado, disponível em Chan e Markov [2004] e Chen et al. [2010]), não nos permite fazer uma comparação rigorosa. No entanto, uma simples análise permite concluir que os nossos tempos são claramente inferiores. Na Tabela 4.5, apresenta-se a comparação com os tempos de execução publicados em Bortfeldt [2013]. Nesta tabela, os tempos da coluna T' resultaram da aplicação de um fator corretivo que reflete as velocidades relativas dos computadores utilizados nos dois conjuntos de testes. O fator corretivo foi calculado com base nos dados disponíveis em www.cpubenchmark.net. Da comparação apresentada podemos concluir que os tempos de execução da heurística Rect-TOPOS são bastante competitivos.

Assim, como seria de esperar, os resultados obtidos com o Rect-TOPOS ficam aquém dos publicados em Bortfeldt [2013], mas em contrapartida os tempos de execução são muito inferiores. Na figura 4.22 apresenta-se a solução obtida, pelo Rect-TOPOS, para a instância PCB500.

A Tabela 4.5 permite também mostrar o grande impacto que o número de tipos de itens tem nos tempos de execução da heurística Rect-TOPOS. Por exemplo, apesar de a instância PCB146 ter mais 46 itens que a instância RP100 o seu tempo de execução é cerca de 10 vezes inferior, já que tem apenas 22 tipos de itens diferentes enquanto a instância RP100 tem 99 tipos diferentes.

Tabela 4.5: Comparação dos tempos de execução entre a heurística Rect-TOPOS e a abordagem apresentada em Bortfeldt [2013].

| Instância | # Rect-TOPOS | # Bortfeldt [2013] | |
|-----------|--------------|--------------------|----------|
| | T_m (s) | # T (s) | # T' (s) |
| APTE | 0.01 | 1.0 | 0.69 |
| AMI33 | 0.84 | 116.0 | 71.37 |
| AMI49 | 1.97 | 1752.0 | 1077.87 |
| RP100 | 9.35 | 59.0 | 36.29 |
| PCB146 | 0.95 | 2250.0 | 1384.25 |
| RP200 | 13.2 | 14.0 | 8.61 |
| PCB500 | 221.0 | 554.0 | 340.83 |

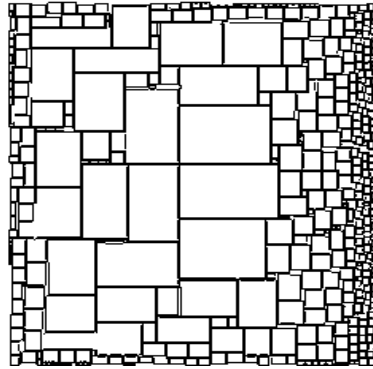


Figura 4.22: Solução obtida para a instância PCB500

4.2.4 Comentários e Conclusões

Nesta secção apresentou-se uma heurística construtiva, Rect-TOPOS, para o problema de PP de minimização de área. Trata-se de uma heurística rápida que é capaz de lidar facilmente com todo o tipo de polígonos ortogonais. Esta heurística usa vários critérios para escolher o próximo item a posicionar, a sua orientação e o melhor ponto de posicionamento. A qualidade das soluções obtidas revelou-se bastante satisfatória dado tratar-se de uma simples heurística com reduzidos tempos de execução. Estas características sugerem, como desenvolvimentos futuros, a incorporação da heurística Rect-TOPOS numa abordagem baseada em pesquisa local. Nesta abordagem seria possível, à custa de um aumento do tempo de execução, melhorar os já bons resultados alcançados pelo Rect-TOPOS em situações onde tal fosse necessário. O tempo de execução para o cálculo do invólucro de posicionamento entre pares de itens sugere o desenvolvimento de um método específico para lidar com itens ortogonais.

Capítulo 5

Abordagens Híbridas para Problemas de Posicionamento de Minimização da Área

Neste capítulo serão abordados métodos heurísticos combinados com modelos matemáticos. Métodos heurísticos e modelos matemáticos foram, durante muitos anos, usados separadamente. Os modelos matemáticos garantem a solução ótima de um problema mas são, principalmente para instâncias de média/grande dimensão, computacionalmente dispendiosos. Para estas instâncias é habitual o recurso a métodos heurísticos na sua resolução. Estes métodos conseguem obter soluções de qualidade inferior num tempo computacional reduzido.

Os métodos heurísticos e os modelos matemáticos podem ser combinados de diferentes formas, por exemplo, os métodos heurísticos podem ser usados dentro de algoritmos *branch-and-bound* para a obtenção de bons limites superiores. Problemas relaxados podem ser usados para guiar a pesquisa do espaço de soluções de uma meta-heurística, uma vez que a solução ótima de uma relaxação de um dado problema indica muitas vezes em que zona do espaço de soluções original podem ser encontradas boas soluções. Os modelos matemáticos podem também ser usados como decodificadores de soluções resultantes de perturbações realizadas em meta-heurística, principalmente quando as pesquisas são realizadas em sequências. Apenas nos

últimos anos os investigadores começaram a reconhecer as vantagens e potencialidades resultantes da combinação destes dois métodos dando origem a abordagens híbridas. O caso particular da combinação de modelos matemáticos e meta-heurísticas também é conhecido por *Matheuristics*. No trabalho de Jourdan et al. [2009] é referido que estes métodos permitem obter bons resultados pois permitem explorar simultaneamente as vantagens dos modelos matemáticos e das meta-heurísticas. Neste trabalho é proposta uma classificação de métodos híbridos que combinem heurísticas com métodos exatos e uma nomenclatura dos métodos usados para esta combinação. A taxonomia apresentada baseia-se no trabalho de Talbi [2002] que, para além de apresentar abordagens que hibridizam meta-heurísticas e modelos matemáticos, faz igualmente referência a abordagens que combinam meta-heurísticas com meta-heurísticas. Uma visão global acerca destes métodos pode ser encontrada em Dumitrescu e Stützle [2003], Feltl e Raidl [2004], Puchinger e Raidl [2005], Raidl [2006] e em Raidl e Puchinger [2008]. Uma classificação geral das abordagens existentes, que combinam modelos matemáticos e meta-heurísticas, é realizada em Puchinger e Raidl [2005]. Estes autores distinguem duas classificações principais resultantes desta hibridização, também representadas na figura 5.1. Uma das combinações é colaborativa, significando que há troca de informação entre os dois métodos, apesar de não fazerem parte integrante um do outro (os dois métodos podem ser realizados sequencialmente, interligados ou em paralelo). Outra combinação é integrativa onde, tal com o nome indica, um dos métodos é parte integrante do outro.

Em PP este tipo de abordagem foi usado por Bennell e Dowsland [2001] onde é combinada a meta-heurística Pesquisa Tabu com modelos matemáticos de compactação e separação para o Problema de *Cutting Stock* com itens irregulares. Mais tarde, uma abordagem híbrida que combina os modelos matemáticos propostos pelos autores anteriores com a meta-heurística pesquisa por arrefecimento simulado foi proposta por Gomes e Oliveira [2006] para o PP de itens irregulares numa tira de altura limitada de comprimento infinito. Como abordagem ao Problema da Mochila unidimensional, Plateau et al. [2002] combinam algoritmos de pontos interiores e meta-heurísticas. Fischetti e Lodi [2003] propuseram uma combinação entre métodos baseados em pesquisa local e abordagens baseadas em programação linear inteira mista para problemas inteiros mistos genéricos. A ideia geral consiste em

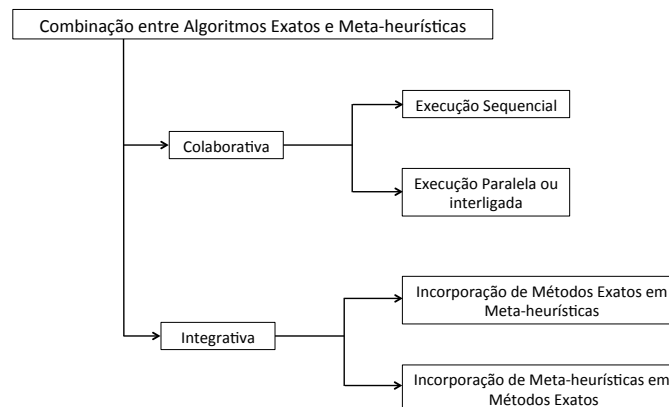


Figura 5.1: Classificação resultante da combinação entre modelos matemáticos e meta-heurísticas (adaptado de Puchinger e Raidl [2005])

fixar um número de variáveis, simplificando o problema, sem perder a possibilidade de obter boas soluções admissíveis. No trabalho de Ibaraki et al. [2008] é feita referência a PP que usam como abordagem métodos híbridos que combinam pesquisa local com técnicas de programação linear, não linear e/ou programação dinâmica. No trabalho de Ibaraki et al. [2008] são referidas três variantes que diferem no conjunto de itens para posicionar: uma considera retângulos com dimensões fixas; numa segunda variante são considerados itens com formas retangulares, no entanto, diferem dos itens da variante anterior por terem dimensões variáveis e área fixa; por último são considerados itens irregulares para posicionamento. Estes autores referem que os resultados dos testes computacionais realizados, pelos diversos autores que usam esta abordagem, indicam que a programação matemática é uma ferramenta vital para melhorar algoritmos meta-heurísticos.

Para lidar com o Problema da Mochila unidimensional, Pisinger [1995] usou relaxação contínua e a propriedade que a solução inteira de um problema envolve a mudança de variáveis próximas à solução associada a um item denominado por item crítico.

Claro que uma das questões fundamentais destes métodos híbridos é como fazer a combinação entre eles de modo a obter benefícios que são inerentes às duas abordagens que compõem o método híbrido.

Neste capítulo apresentam-se abordagens que combinam modelos matemáticos com métodos heurísticos, figura 5.2, para PP de minimização

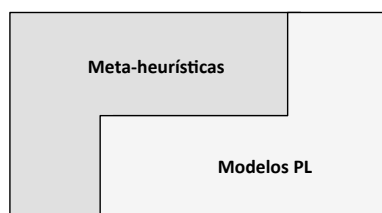


Figura 5.2: Hibridização entre modelos matemáticos e meta heurísticas

de área. A principal motivação para o desenvolvimento destas abordagens relaciona-se com o facto dos modelos apresentados para este problema não serem resolúveis em tempo computacional baixo, exceto para instâncias de pequena dimensão tal como foi visto no capítulo 3. No entanto, é também importante tirar partido do uso de modelos matemáticos, uma vez que conseguem lidar com problemas como um todo.

Na primeira secção deste capítulo apresenta-se uma abordagem híbrida entre a meta-heurística ILS e dois modelos matemáticos, de compactação e separação, que permitem a obtenção de padrões admissíveis e ótimos locais, respetivamente. Trata-se, segundo a classificação apresentada de uma combinação integrativa onde os métodos exatos são inseridos na meta-heurística. Na segunda secção, apresenta-se uma heurística construtiva que se baseia no posicionamento de grupos de itens, com o auxílio de modelos matemáticos.

5.1 Abordagem Híbrida Baseada em Pesquisa sobre Soluções Completas

Nesta secção, apresenta-se uma abordagem híbrida de pesquisa local para o PP de minimização de área. Os algoritmos híbridos podem, tal como já referido anteriormente, combinar meta-heurísticas com modelos matemáticos inseridos na estrutura de vizinhança. Esta abordagem foi usada por Gomes e Oliveira [2006] para a resolução do problema de posicionamento de itens com formas irregulares. Baseia-se numa estrutura de vizinhança que utiliza modelos matemáticos para compactar padrões e separar os itens no caso de existência de sobreposições e usa, para a pesquisa do espaço de soluções, a meta-heurística *Iterated Local Search* (ILS). Os modelos matemáticos são aplicados de forma coordenada. O modelo de compactação opera sobre uma solução admissível e é usado para obter soluções que são ótimos locais, isto

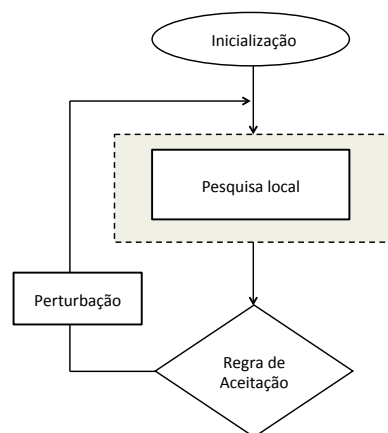


Figura 5.3: Estrutura da abordagem híbrida ILS+Compactação/Separação

é, soluções compactas e admissíveis. O modelo de separação é usado para obter soluções admissíveis resultantes de soluções não admissíveis. Ambos os modelos são construídos a partir das posições relativas entre todos os pares de itens.

Este trabalho encontra-se publicado no livro de atas do EU/MEeting 2009 ([Oliveira et al., 2009]).

5.1.1 Meta-Heurística ILS

Neste trabalho optou-se pelo uso da meta-heurística ILS para a pesquisa do espaço de soluções combinada com os modelos matemáticos de compactação e separação. A opção de usar os mesmos modelos de Gomes e Oliveira [2006] permite-nos lidar com polígonos ortogonais facilmente uma vez que originalmente foram construídos para lidarem com polígonos não convexos. A meta-heurística ILS, proposta por Lourenço et al. [2003], é um algoritmo iterativo que é composto pelos seguintes componentes principais: inicialização, pesquisa local, regra de aceitação e perturbação. No processo de inicialização é obtida uma solução para iniciar o processo. Em cada iteração do ILS é aplicada pesquisa local à solução corrente. De seguida, é usada a regra de aceitação para atualizar ou não a solução corrente, pela solução obtida pela pesquisa local, e o movimento de perturbação é usado para a obtenção de uma nova solução corrente, figura 5.3.

A solução inicial é obtida posicionando os itens numa linha onde a or-

dem dos itens é completamente aleatória, isto é, obtém-se um padrão onde os itens são posicionados aleatoriamente num retângulo suficiente grande para que sejam posicionados numa única linha. Posteriormente são aplicados os modelos matemáticos de compactação e separação para obter padrões compactos e admissíveis. Na pesquisa local, é necessário introduzir pequenas alterações entre as posições relativas dos itens para a obtenção de padrões com diferentes relações entre pares de itens. Deve notar-se que, contrariamente ao que acontece nas maioria das abordagens existentes, para resolver este problema, que usam sequências de itens para a realização de movimentos, esta abordagem opera diretamente sobre padrões. Para a geração de um novo vizinho é introduzida uma pequena alteração no padrão, designada por movimento para gerar um novo vizinho, que é obtida trocando as posições de dois itens no padrão. Usualmente, esta pequena troca de posições de itens resulta em sobreposições que são removidas aplicando o modelo de separação. Subsequentemente é aplicado o modelo de compactação para compactar o padrão. A grande diferença entre estes dois modelos reside na função objetivo: no modelo de compactação é a área do padrão enquanto que no modelo de separação é a medida de sobreposição dos itens. Uma vez que o tamanho de uma vizinhança usada na pesquisa local cresce de forma polinomial com o aumento do número de itens, apenas é pesquisada uma vizinhança parcial. Usaram-se duas regras de aceitação distintas: uma que atualiza sempre a solução corrente e outra que apenas atualiza se a nova solução for melhor do que a solução corrente. O motivo para a escolha da primeira regra é que esta permite guiar a pesquisa para outra região do espaço de soluções (i.e para diversificar a pesquisa), com a segunda regra pretende-se intensificar a pesquisa em torno da melhor solução encontrada até ao momento. A perturbação altera uma percentagem de itens do padrão atual para novas posições geradas aleatoriamente. Na figura 5.4 apresenta-se a estrutura detalhada da abordagem usada, realçando-se o bloco denominado *Otimização local*, onde são chamados de forma coordenada os modelos de compactação e separação.

5.1.2 Modelos de Compactação e Separação

Na abordagem híbrida proposta, um padrão é analisado e estabelecem-se relações de posicionamento entre pares de itens, sendo de seguida aplicados os modelos de compactação e separação, figura 5.5. Assim, a utilização destes dois modelos de forma coordenada permite a obtenção de padrões

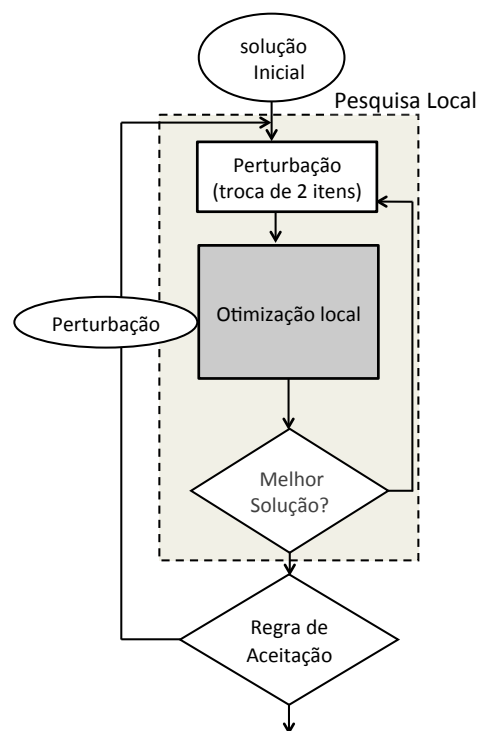


Figura 5.4: Estrutura detalhada da abordagem híbrida ILS+Compactação/Separação

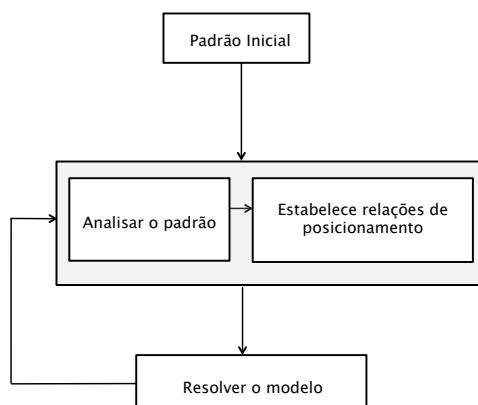


Figura 5.5: Otimização local

compactos e admissíveis a partir de um padrão inicial que pode ser não admissível. Usualmente, quando é realizado um movimento de troca de dois itens, surgem sobreposições entre pares de itens, resultando em padrões não admissíveis. O modelo de separação, tal como já foi referido anteriormente, permite a remoção de situações de sobreposição entre itens posicionados, figura 5.6(a) e, conseqüentemente, a obtenção de padrões admissíveis. Durante este processo não é possível existir um aumento de situações de não admissibilidade e os movimentos entre itens são contínuos. O modelo de compactação parte de soluções admissíveis e permite obter padrões compactos, aumentando o aproveitamento do padrão. Neste modelo um conjunto de movimentos coordenados e contínuos são aplicados aos itens resultando em padrões que são ótimos locais. No padrão obtido, depois de aplicado o modelo de compactação, são mantidas as posições relativas entre pares de itens apesar do posicionamento exato dos itens poder ser completamente distinto do existente no padrão inicial, figura 5.6(b).

A estrutura de vizinhança usada consiste na operação de troca entre as posições de dois itens selecionados aleatoriamente. A figura 5.7 ilustra um exemplo desta operação e da obtenção de padrões admissíveis e compactos obtidos pela aplicação dos modelos de separação e compactação.

5.1.3 Testes Computacionais

Os itens usados para os testes desta subsecção são retangulares. Foram usadas apenas as instâncias com número total de itens inferior a 50, do

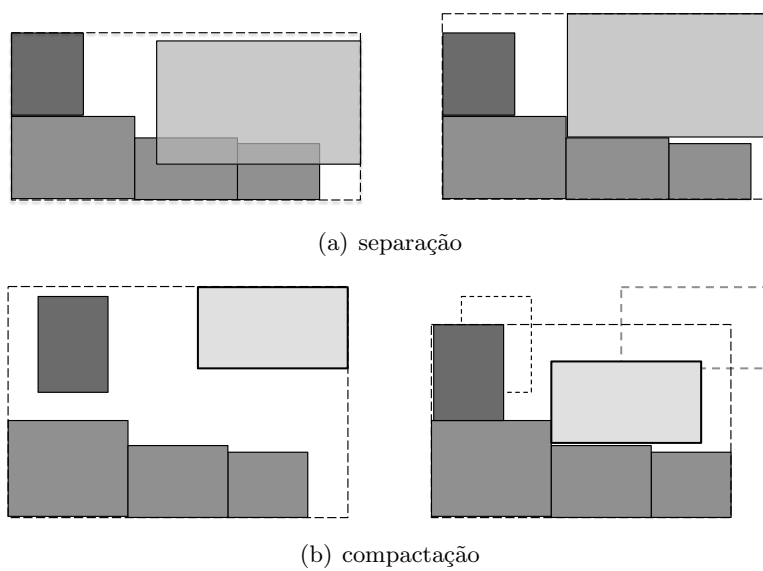


Figura 5.6: Compactação e separação de itens

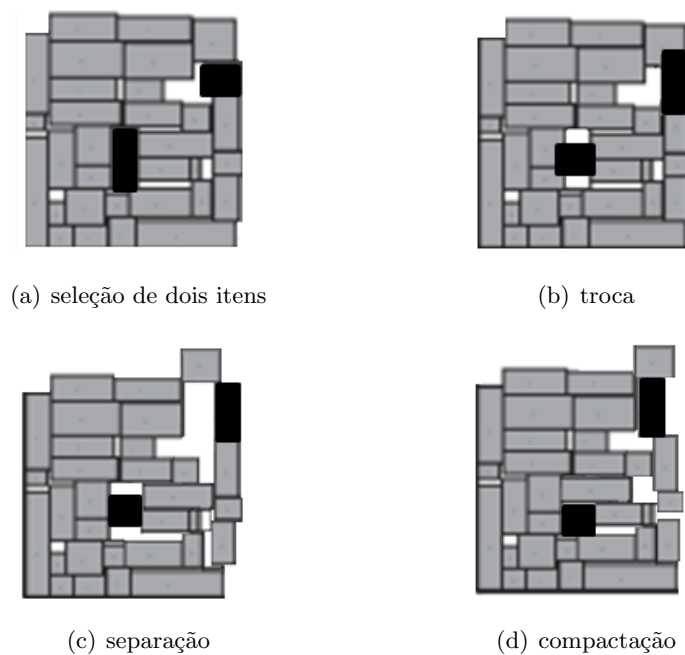


Figura 5.7: Exemplo de uma operação de troca de posições entre dois itens e da aplicação dos modelos de compactação e separação

conjunto de referência MCNC¹ com origem no projeto de circuitos VLSI.

Foram implementadas e avaliadas duas versões diferentes da abordagem híbrida: ILS_CS e ILS_BS. A versão ILS_CS baseia-se na primeira regra de aceitação descrita anteriormente (onde a solução corrente é sempre atualizada), enquanto que a versão ILS_BS baseia-se na segunda regra (a solução corrente apenas é atualizada se a nova solução for melhor). Para melhor se realçar o comportamento das duas versões (ILS_CS diversifica a pesquisa através de movimentos para diferentes soluções, enquanto ILS_BS intensifica a pesquisa sempre à volta da melhor solução), foram usadas na perturbação duas percentagens diferentes: 10% dos retângulos na versão ILS_CS e 5% na versão ILS_BS. Os restantes parâmetros usados são comuns às duas versões: 20 iterações no ILS e a vizinhança na pesquisa local é limitada a 200 trocas, no máximo, ou a 25% do número total de trocas possíveis (estes valores foram fixados após a realização de alguns testes preliminares).

Os testes computacionais foram realizados numa estação de trabalho Linux equipada com duplo processador Intel XEON Dual Core 5160, 3GHz e foi usado o ILOG Cplex 10.1 para resolver os modelos de programação linear. Os resultados obtidos apresentam-se na tabela 5.1. Nesta tabela apresenta-se, para cada instância, a melhor solução, a pior solução, a média das vinte corridas e o tempo médio de execução para uma corrida, assim como os melhores resultados conhecidos na literatura.

5.1.4 Comentários e Conclusões

Os resultados obtidos pelas duas versões, da abordagem híbrida, propostas foram muito consistentes. Em ambas as versões, os resultados mostraram diferenças, entre as melhores soluções e as médias das melhores soluções, de cerca de 1% para quatro instâncias. Apenas na instância HP esta diferença foi superior a 2%. As duas versões usadas mostraram resultados similares. Na tabela 5.1 pode-se observar que, entre as duas versões, a diferença entre a média, a melhor e a pior solução é sempre inferior a 0.5% com uma pequena vantagem na média das soluções para a versão ILS_BS e na melhor solução para ILS_CS. Tal como era esperado, as duas versões apresentam tempos de execução semelhantes, uma vez que ambos partilham a mesma estrutura. Para a média das soluções, a diferença é inferior a 0.5% para a instância AMI33, cerca de 3% para a instância HP e cerca de 2% para as restantes

¹<http://vlsicad.eecs.umich.edu/BK/MCNCbench/HARD/>

Tabela 5.1: Resultados obtidos pela abordagem híbrida (20 corridas)

| Instância | ILS_CS | | | | Melhor Resultado | |
|-----------|-----------|------------|----------|-----------------|------------------|-----------------------------|
| | Média (%) | Melhor (%) | Pior (%) | Tempo Médio (s) | Aprov. (%) | Tempo s Ref. |
| APTE | 97.59 | 98.41 | 93.62 | 8.7 | 99.2 | 1.0 Bortfeldt [2013] |
| XEROX | 96.42 | 97.57 | 94.52 | 13.6 | 97.7 | 9812.0 Chan e Markov [2004] |
| HP | 95.41 | 97.77 | 92.25 | 12.5 | 98.7 | 891.0 Chan e Markov [2004] |
| AMI33 | 95.92 | 96.77 | 95.18 | 699.9 | 99.0 | 116.0 Bortfeldt [2013] |
| AMI49 | 95.95 | 96.98 | 95.18 | 2750.4 | 98.6 | 1752.0 Bortfeldt [2013] |

| Instância | ILS_BS | | | | Melhor Resultado | |
|-----------|-----------|------------|----------|-----------------|------------------|-----------------------------|
| | Média (%) | Melhor (%) | Pior (%) | Tempo Médio (s) | Aprov. (%) | Tempo s Ref. |
| APTE | 97.56 | 98.41 | 93.62 | 9.4 | 99.2 | 1.0 Bortfeldt [2013] |
| XEROX | 96.57 | 97.58 | 95.25 | 14.2 | 97.7 | 9812.0 Chan e Markov [2004] |
| HP | 95.71 | 97.77 | 92.25 | 12.1 | 98.7 | 891.0 Chan e Markov [2004] |
| AMI33 | 95.72 | 96.52 | 94.78 | 876.3 | 99.0 | 116.0 Bortfeldt [2013] |
| AMI49 | 95.98 | 96.71 | 95.10 | 2579.1 | 98.6 | 1752.0 Bortfeldt [2013] |

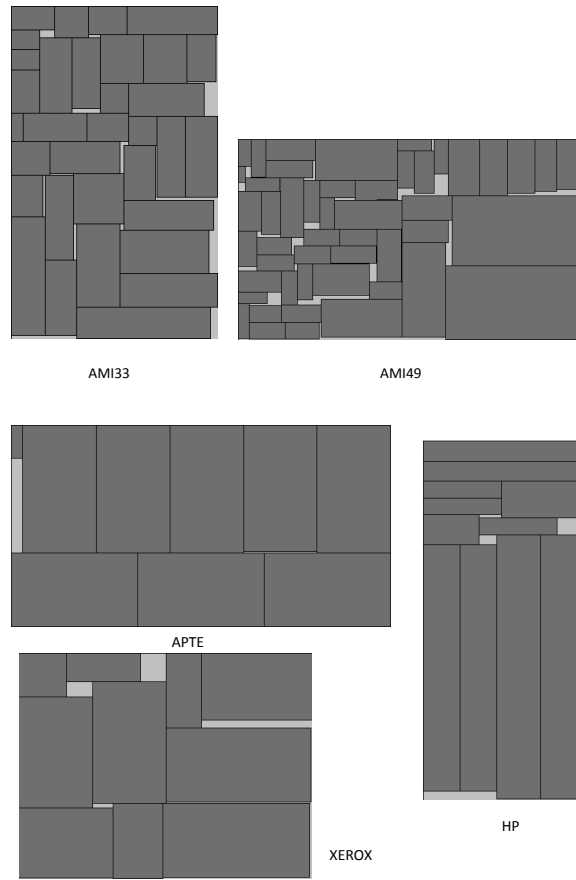


Figura 5.8: Exemplos de padrões obtidos através da abordagem ILS+Compactação/Separação

instâncias. Os tempos de execução apresentados na tabela 5.1 sugerem que esta abordagem apenas consegue lidar, em tempo útil, com instâncias de pequena/média dimensão. Na figura 5.8 apresentam-se exemplos de padrões obtidos pela abordagem híbrida ILS+Compactação/Separação.

5.2 Abordagem Híbrida Baseada na Construção de Padrões

Com base na versão linearizada do modelo apresentado na secção 3.3 do capítulo 3, para PP de minimização de área, desenvolveu-se uma abordagem híbrida iterativa de construção de padrões. A abordagem híbrida proposta

usa regras heurísticas para seleccionar os itens a posicionar em cada iteração e modelos matemáticos para resolver iterativamente os subproblemas, tornando assim possível a resolução de instâncias complexas e de grande dimensão.

5.2.1 Heurística Construtiva Híbrida

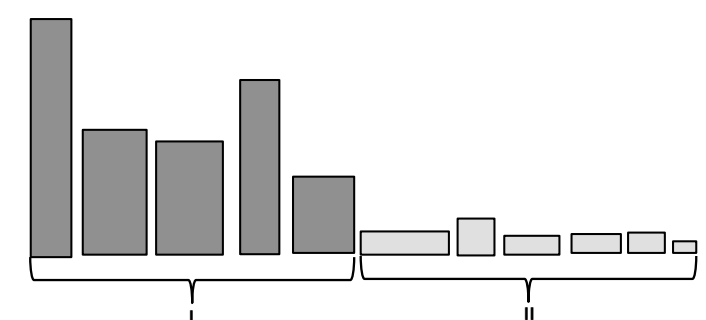
Nesta subsecção apresenta-se uma abordagem híbrida que constrói soluções através de um processo iterativo. Cada subconjunto de itens a posicionar é obtido através de regras heurísticas que se baseiam no tamanho dos itens. Em cada iteração, as posições relativas entre pares de itens posicionados são mantidas fixas, enquanto que os novos itens a posicionar têm posições relativas livres. A principal ideia é posicionar os itens maiores, em primeiro lugar, e usar os itens mais pequenos para preencher os espaços vazios que resultam do posicionamento dos itens maiores.

O processo inicia sempre com uma solução parcial nula e, a cada iteração, um item ou um conjunto de itens são adicionados à solução parcial. O processo termina quando for construída uma solução que contenha todos os itens. Com a heurística proposta pretende-se obter boas soluções, num tempo computacional baixo, mesmo para instâncias de grande dimensão. Tal como foi referido anteriormente, o número de variáveis binárias existentes no modelo apresentado, que dizem respeito às posições relativas entre pares de itens, cresce de forma polinomial com o aumento do número de itens, sendo este número de variáveis muito significativo. Dadas as dificuldades em resolver o modelo na sua globalidade, em tempo útil, é natural o uso de um método heurístico que vá adicionando os itens a uma solução parcial (conjunto de itens posicionados anteriormente), em vez de os posicionar todos de uma só vez.

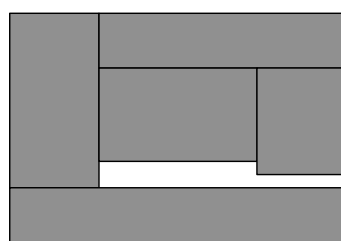
Tal como já foi referido, o modelo matemático apresentado na secção 3.3 do capítulo 3 não resolve, em tempo útil, instâncias de média e grande dimensão. No entanto, é importante tirar partido das vantagens resultantes do modelo matemático na resolução de PP, nomeadamente a resolução de problemas como um todo. Nos problemas reais que, na maiorias das vezes, são de dimensão elevada é importante reduzir o tempo computacional e, simultaneamente, obter soluções de boa qualidade. A ideia geral da abordagem proposta, denominada HCH, consiste na decomposição de problemas em pequenos subproblemas, permitindo diminuir a sua complexidade, de

forma a ser possível a resolução de cada subproblema através do modelo matemático. Como se trata de um processo iterativo, o modelo matemático vai sendo construído em cada iteração por adição de novas variáveis e restrições. Em vez de serem posicionados todos os itens, em cada iteração da heurística construtiva, é posicionado um conjunto de itens que é adicionado ao conjunto de itens anteriormente posicionados (solução parcial). A primeira iteração distingue-se das restantes por ser possível adicionar um maior número de itens ao modelo. Nas iterações seguintes, adicionam-se ao modelo da iteração anterior as variáveis e restrições relativas aos novos itens que se pretendem adicionar e, para além disso, as variáveis binárias referentes às posições relativas entre pares de itens, que constituem a solução parcial, são fixadas de acordo com o posicionamento destes itens. Fixar as variáveis binárias consiste na atribuição de um valor particular (zero ou um) às variáveis r_{ij} e u_{ij} . Por exemplo, se o item i está posicionado completamente à esquerda do item j então a variável r_{ij} terá o valor fixo igual a zero. Na figura 5.9 está ilustrado um exemplo de uma solução obtida através da heurística construtiva híbrida. Neste exemplo, um conjunto de itens de vários tamanhos está disponível para posicionamento; estes itens são ordenados segundo um determinado critério; na figura 5.9(a) os itens estão ordenados por ordem decrescente de área; de seguida é feita uma seleção de itens, sendo escolhidos os itens de maior área para posicionar em primeiro lugar (representados na figura por I), o posicionamento é obtido através da resolução do modelo matemático, figura 5.9(b); na iteração seguinte, um novo conjunto de itens (conjunto II) é selecionado para adicionar à solução parcial obtida, onde as posições relativas entre pares de itens da solução parcial, que resultou do posicionamento dos itens do conjunto I, se mantêm inalteradas; o posicionamento dos itens é feito através da resolução do modelo matemático. Não havendo mais itens para posicionar é obtida a solução final, figura 5.9(c).

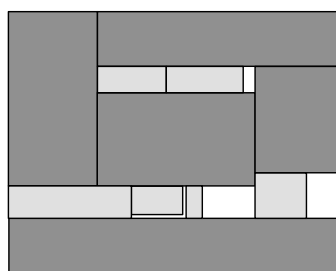
No fluxograma da figura 5.10 está representada a sequência de passos da abordagem proposta. É sabido que os itens maiores (maior área, maior perímetro, maior largura, etc.) têm mais influência no valor da função objetivo do que os itens mais pequenos, que resulta, muitas vezes, das dimensões do padrão serem determinadas pelo posicionamento dos itens maiores. Por este motivo, torna-se importante que os itens maiores sejam posicionados no início, pois é muitas vezes possível o preenchimento de espaços vazios, re-



(a) seleção de itens para posicionar



(b) solução da fase I



(c) solução da fase II

Figura 5.9: Ilustração do funcionamento da heurística construtiva

sultantes do posicionamento dos itens maiores, com os itens mais pequenos. Consequentemente, é necessário uma ordenação dos itens, por ordem decrescente, segundo um critério que pode ser, por exemplo, área, perímetro, largura, etc. Depois desta ordenação, os primeiros k_0 itens são escolhidos para serem posicionados em primeiro lugar. O modelo matemático apresentado na secção 3.3 do capítulo 3 é construído iterativamente e, consecutivamente, resolvido com as variáveis e restrições relativas aos itens que se pretendem posicionar. Em cada iteração, a solução do modelo é um conjunto de pontos de posicionamento que formam uma solução parcial. Depois de se obter uma solução parcial, e se ainda houver itens para posicionar, são escolhidos k itens para adicionar à solução parcial anterior. Ao modelo anteriormente resolvido são adicionadas restrições e variáveis respeitantes a estes itens. Este aumento de restrições e variáveis torna o modelo mais complexo à medida que o número de iterações aumenta. Para diminuir esta complexidade, as posições relativas entre pares de itens, que constituem a solução parcial, permanecem inalteradas. Para fixar as relações geométricas entre pares de itens foram usadas três variantes que serão explicadas mais à frente. Este processo iterativo continua até que todos os itens estejam posicionados. A solução parcial da última iteração é a solução final do modelo.

Em relação ao valor de k_0 da primeira iteração, deve notar-se que à medida que k_0 aumenta torna-se mais difícil resolver o modelo até à optimalidade. Por outro lado, pretende-se encontrar um bom posicionamento relativo entre itens grandes, não sendo, no entanto, necessário obter a solução ótima para esta solução parcial, uma vez que o ponto de posicionamento de cada item poderá não ser o mesmo até à obtenção da solução final. Nas iterações subsequentes, o valor de k não pode ser muito elevado pois, à medida que o número de itens posicionados aumenta, o modelo torna-se mais complexo, resultante do aumento do número de variáveis e restrições mas, em contrapartida, torna-se mais fácil pois, com a diminuição do tamanho dos itens, é possível preencher espaços vazios. Esta possibilidade de preencher os espaços vazios é uma das grandes vantagens desta heurística. Tal como para os parâmetros k_0 e k poderá ser também necessário fazer pequenos ajustes aos valores do *gap* e do tempo limite de resolução do modelo em cada iteração. Um bom ajuste do *gap* permite obter boas soluções num tempo computacional rápido. Quanto mais pequeno for este valor melhor será a solução obtida mas, em contrapartida, torna a sua obtenção mais

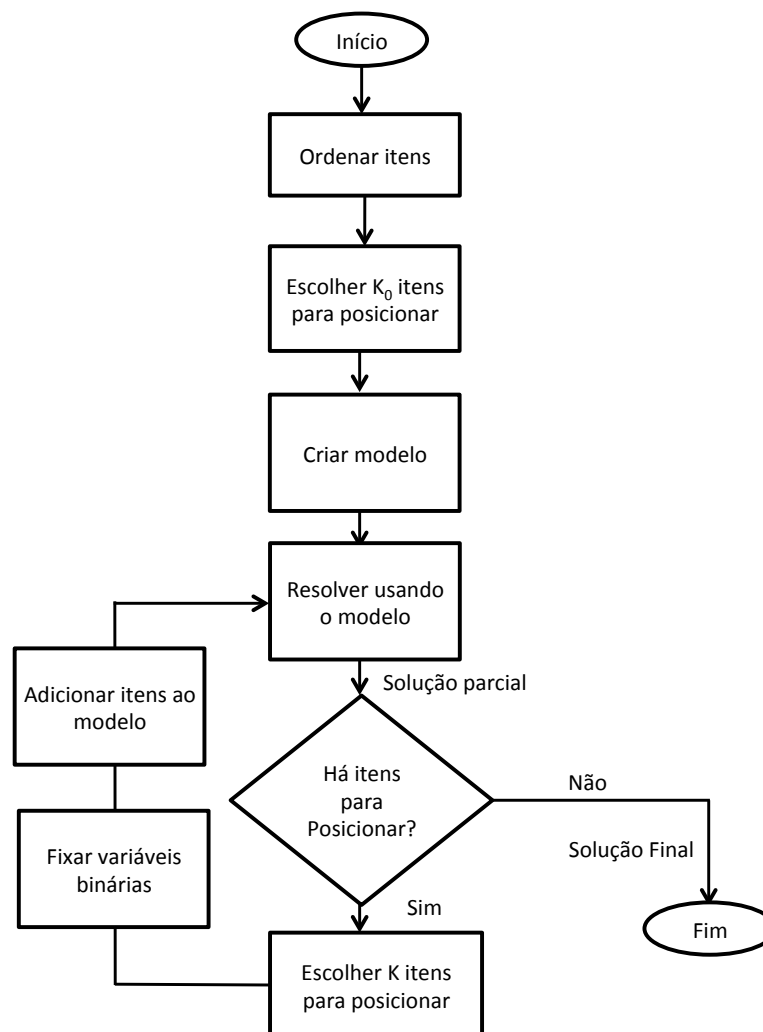


Figura 5.10: Fluxograma da heurística construtiva híbrida com modelos matemáticos

demorada. Por outro lado, o tempo limite deve ser ajustado de forma a que, combinado com o *gap*, permita que o modelo pare em casos onde não é possível atingir em tempo útil o *gap* escolhido. Esta situação pode ocorrer, por exemplo, quando, com os itens existentes, não for possível obter um bom encaixe resultando num caso de posicionamento mais complexo e computacionalmente mais demorado. Estes dois parâmetros devem ser configurados de forma adequada para permitir um bom funcionamento da heurística e, consequentemente, obter soluções de boa qualidade num tempo computacional aceitável.

Para fixar as posições relativas entre pares de itens, tal como foi referido anteriormente, foram consideradas três variantes desta heurística: uma onde as posições relativas entre pares de itens pertencentes à solução parcial estão fixas (V_1), outra onde os centros destes itens variam num determinado intervalo (V_2) e, finalmente, uma terceira variante onde os centros dos itens da solução parcial se mantêm inalterados (V_3). Apesar de ser computacionalmente mais dispendiosa, a primeira variante é mais adequada à resolução de modelos com menos de 49 itens, uma vez que é a mais flexível pois, com a possibilidade de afastamento dos itens posicionados, permite maior liberdade ao posicionamento dos itens que vão ser adicionados à solução parcial. A segunda versão aplica-se à resolução de modelos com número de itens entre 49 e 200 e consiste em permitir pequenas variações nos pontos de posicionamento de itens posicionados em iterações anteriores. A terceira variante é mais indicada para modelos com mais de 200 itens e consiste na fixação dos pontos de posicionamento do conjunto de itens da solução parcial, sendo a mais rígida das três variantes mas, por outro lado, a menos demorada.

Com a primeira variante pretende-se manter as posições relativas entre pares de itens que pertençam à solução parcial permitindo, em iterações seguintes, posicionar, se possível, itens no meio destes, preenchendo espaços vazios, não resultando num aumento do valor da função objetivo. Manter as posições relativas entre pares de itens significa que, se um item está à esquerda de outro, por exemplo, a posição relativa entre eles será sempre esta, podendo o seu afastamento ser maior ou menor. Com esta estratégia, o número de variáveis binárias, relativamente ao modelo global, não cresce de forma tão acentuada, apenas sendo necessário considerar variáveis binárias entre os pares de itens que vão sendo adicionados e, variáveis binárias entre estes itens e os que já estão posicionados. Não é necessário considerar

variáveis binárias entre pares de itens já posicionados, uma vez que estas variáveis estão fixas, isto é, ou são iguais a zero ou a um. Por exemplo, se uma solução parcial tem seis itens posicionados e, pretendendo-se adicionar três itens, obtêm-se 48 variáveis binárias para as posições relativas entre pares de itens em vez das 90 resultantes do modelo completo. Esta estratégia permite resolver o modelo em tempo útil.

A segunda variante permite fazer o mesmo que a versão anterior mas com maior rigidez, não dando tanta liberdade de movimentação aos itens que formam uma solução parcial. Depois de posicionado um conjunto de itens que formam uma solução parcial, nas iterações seguintes o centro destes itens pode variar apenas num intervalo $x_i^c \in [x_i^c - \Delta, x_i^c + \Delta]$ e $y_i^c \in [y_i^c - \Delta, y_i^c + \Delta]$. A posição relativa entre pares de itens permanece inalterada de forma indireta, isto é, algumas variáveis binárias são automaticamente eliminadas porque conduzem a situações onde não existem soluções admissíveis. O objetivo é permitir um determinado afastamento entre itens de forma a permitir pequenos ajustes na solução parcial. O valor do parâmetro Δ deverá ter em atenção as dimensões dos itens das instâncias que se pretendem resolver de forma a que seja possível o posicionamento de itens mais pequenos entre itens posicionados em iterações anteriores.

A terceira variante surge da necessidade de resolver modelos com mais de 200 itens. Para estas situações, a redução do número de variáveis binárias pelo processo anteriormente descrito não é suficiente, para a sua resolução num tempo aceitável. Este problema resulta do crescimento da solução parcial pois é necessário considerar variáveis binárias entre todos os itens que se vão posicionar e todos os itens já posicionados. Por este motivo, nesta variante optou-se por fixar efetivamente o ponto de posicionamento de cada item da solução parcial, não possibilitando quaisquer mudanças nos pontos de posicionamentos até à obtenção da solução final. Esta variante é a mais rígida mas, em contrapartida, o número de variáveis binárias é mais reduzido do que nas versões anteriores.

Quando os itens a posicionar são ortogonais é necessário garantir que se um dos itens retangulares que forma este polígono é escolhido para ser adicionado à solução parcial todos os outros também têm que ser selecionados. Note-se que o modelo, quando lida com itens ortogonais, não é tão demorado como no posicionamento separado do mesmo número de retângulos que constituem o item ortogonal, uma vez que algumas variáveis, para além da

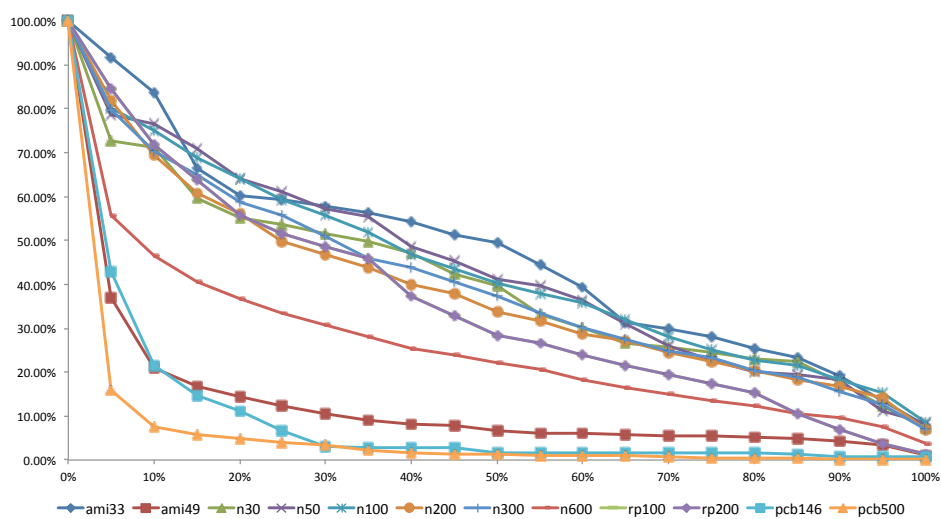
posição relativa, estão fixas em relação a outras.

5.2.2 Análise às Instâncias

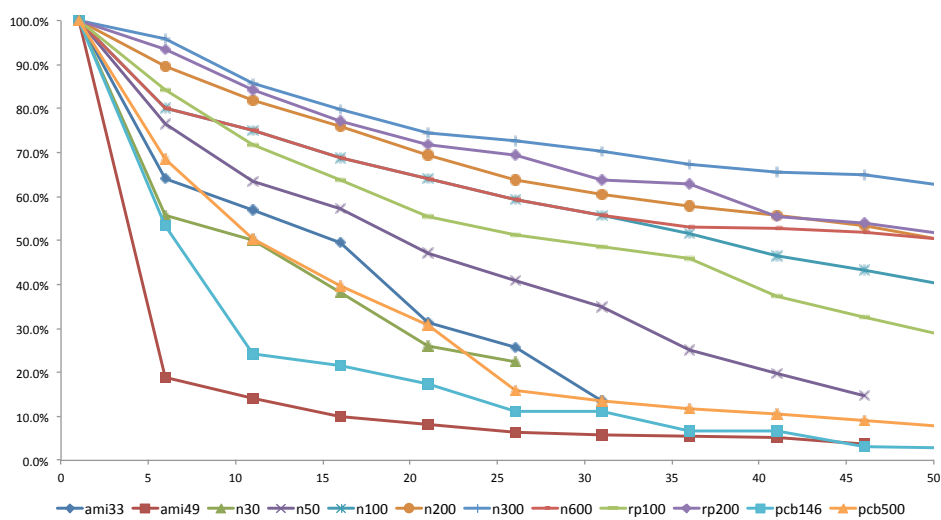
Tal como foi referido anteriormente, é importante ter alguma informação acerca das instâncias que se pretendem resolver, nomeadamente no que se refere aos tamanhos dos itens. O tamanho de um item pode ser a sua área, perímetro, altura, largura, etc. A importância desta análise reside principalmente na seleção dos itens que formam a primeira solução parcial, isto é, no valor de k_0 e na escolha do valor de Δ .

Nas figuras 5.11(a) e 5.11(b) estão representados dois gráficos que apenas diferem no eixo do x . Ambos os gráficos representam a variação das áreas dos itens, ordenadas por ordem decrescente. Do gráfico da figura 5.11(a) representa k -ésimo percentil de valores da área dos itens, onde k está representado no eixo do x . Pode usar-se a informação representada neste gráfico para estabelecer um limite para o número de itens a usar na primeira iteração da heurística construtiva híbrida. O segundo gráfico apresenta a variação das dimensões dos itens até ao quinquagésimo item. Da figura 5.11(a) pode verificar-se que existem essencialmente três grupos distintos de instâncias. Um dos grupos, constituído pelas instâncias AMI49, PCB146 e PCB500, tem um grande número de itens muito pequenos; um segundo grupo, composto pela instância N600, com itens médios e pequenos; e um grupo onde as instâncias têm itens médios, pertencem a este grupo as instâncias AMI33, N30, N50, N100, N200, N300, RP100 e RP200. Nas instâncias pertencentes ao primeiro grupo a possibilidade de preencher os espaços vazios é maior dado o elevado número de itens de pequena dimensão. O gráfico da figura 5.11(b) pode concluir-se que a escolha de k_0 difere de instância para instância.

Na tabela 5.2 apresentam-se algumas estatísticas relativas às dimensões de cada instância, como: a menor dimensão, o primeiro quartil, a média, a mediana, o terceiro quartil e a maior dimensão. Para além destes valores apresenta-se ainda o valor de Δ , que, para cada instância, corresponde a metade da 25^a maior dimensão de todas as dimensões da instância. Nos testes computacionais que se irão apresentar de seguida será considerado, para cada instância o valor de Δ apresentado nesta tabela.



(a)



(b)

Figura 5.11: Variação da área dos itens em cada instância

Tabela 5.2: Análise estatística e valor do Δ

| Instância | Min. | Q1 | Média | Mediana | Q3 | Máx. | Δ |
|-----------|------|-------|--------|---------|--------|------|----------|
| N30 | 33 | 55.75 | 83.77 | 82.5 | 109 | 134 | 48 |
| N50 | 25 | 42 | 62.65 | 65 | 82 | 96 | 41 |
| ami33 | 49 | 119 | 195.47 | 140 | 225.75 | 560 | 91 |
| ami49 | 168 | 406 | 809.57 | 749 | 976.5 | 3234 | 490 |
| N100 | 16 | 30.5 | 42.56 | 42 | 56 | 67 | 31 |
| N200 | 12 | 20 | 29.6 | 30 | 38 | 48 | 23 |
| RP100 | 2 | 30 | 45.54 | 42 | 60.5 | 95 | 38 |
| RP200 | 2 | 30 | 45.54 | 42 | 60.5 | 95 | 43 |
| PCB146 | 160 | 250 | 586.01 | 400 | 640 | 6596 | 610 |
| N300 | 12 | 21 | 30.06 | 30.5 | 39 | 48 | 23 |
| N600 | 12 | 22 | 31.99 | 32.5 | 41 | 67 | 31 |
| PCB500 | 11 | 15 | 38.81 | 27 | 46.25 | 295 | 81 |

5.2.3 Testes Computacionais

Nesta subsecção apresentam-se os testes e experiências computacionais realizados com a heurística construtiva híbrida apresentada na secção 5.2.1. Estes testes estão dividido em duas partes, uma onde as instâncias contêm apenas itens retangulares e outra onde as instâncias contêm, além de retângulos, itens ortogonais.

Itens Retangulares

Os testes computacionais realizados baseiam-se na resolução de três grupos principais de instâncias, que contêm apenas itens retangulares. Um dos grupos contém instâncias de pequena/média dimensão, com um número de itens que varia entre trinta e cinquenta; um segundo grupo com instâncias de dimensão média/grande onde o número total de itens varia entre cem e duzentos e, finalmente, no terceiro conjunto incluem-se instâncias grandes, com um número total de itens entre trezentos e seiscentos. Todas as instâncias têm origem no projeto de circuitos VLSI e podem ser obtidas em <http://vlsicad.eecs.umich.edu/BK/MCNCbench/HARD/>. Na tabela 5.3 apresenta-se, além do nome da instância e do número total de itens, o melhor resultado conhecido para cada uma das instâncias de referência, nomeadamente a percentagem de aproveitamento e o tempo em

Tabela 5.3: Melhores resultados conhecidos

| Instância | #Rect. | Melhor Res. | | |
|-----------|--------|-------------|-----------|----------------------|
| | | Aprov. (%) | Tempo (s) | Ref. |
| N30 | 30 | 97.8 | 1543.0 | Bortfeldt [2013] |
| N50 | 50 | 98.5 | 7.10 | Li et al. [2010] |
| AMI33 | 33 | 99.0 | 116.0 | Bortfeldt [2013] |
| AMI49 | 49 | 98.6 | 1752.0 | Bortfeldt [2013] |
| N100 | 100 | 99.8 | 1915.0 | Bortfeldt [2013] |
| N200 | 200 | 99.0 | 37.0 | Bortfeldt [2013] |
| RP100 | 100 | 99.0 | 59.0 | Bortfeldt [2013] |
| RP200 | 200 | 99.1 | 14.0 | Bortfeldt [2013] |
| PCB146 | 146 | 98.85 | 2250.0 | Bortfeldt [2013] |
| N300 | 300 | 99.0 | 39.0 | Bortfeldt [2013] |
| N600 | 600 | 93.1 | 90.0 | Chan e Markov [2004] |
| PCB500 | 500 | 99.0 | 554.0 | Bortfeldt [2013] |

segundos. É também apresentada a referência ao trabalho onde o melhor resultado foi obtido. No caso de existirem trabalhos com o mesmo valor optou-se por colocar na tabela o que obteve menor tempo de execução.

Tal como já foi referido na secção 5.2.1 existem três variantes possíveis de considerar na heurística construtiva híbrida, todas relacionadas com os itens que formam uma solução parcial: uma onde as variáveis respeitantes às posições relativas entre pares de itens estão fixas; uma onde se dá uma pequena liberdade de movimentação dos centros dos itens (relacionada com o valor Δ); uma onde os centros dos itens da solução parcial estão fixos. Em vez de se considerar cada variante isoladamente optou-se por relacionar a sua variação com o número de itens posicionados até ao momento, em cada iteração. Assim, considerando S_i a soma do número de itens posicionados até à iteração i , temos:

- $0 < S_i \leq 25$ apenas se fixam as posições relativas entre pares de itens;
- $25 < S_i \leq 50$ é permitida uma variação igual a Δ ;
- $50 < S_i \leq 75$ é permitida uma variação igual a $\frac{\Delta}{2}$;
- $75 < S_i \leq 100$ é permitida uma variação igual a $\frac{\Delta}{5}$;
- $100 < S_i \leq 150$ é permitida uma variação igual a $\frac{\Delta}{10}$;

Tabela 5.4: Resultados dos testes computacionais da heurística construtiva híbrida numa configuração mais demorada

| Instância | TL (s) | k_0 | k | Δ | Tempo (s) | W | H | Aprov. (%) |
|-----------|--------|-------|-----|----------|-----------|-------|-------|------------|
| N30 | 60 | 8 | 3 | 48 | 116.215 | 431 | 515 | 93.8 |
| N50 | 60 | 6 | 3 | 41 | 257.993 | 458 | 454 | 95.5 |
| AMI33 | 30 | 6 | 2 | 91 | 15.1153 | 1260 | 966 | 95.0 |
| AMI49 | 60 | 6 | 3 | 490 | 181.543 | 6720 | 5530 | 95.4 |
| N100 | 60 | 6 | 3 | 31 | 1044.62 | 358 | 538 | 93.2 |
| N200 | 60 | 6 | 3 | 23 | 1062.41 | 447 | 437.7 | 89.8 |
| RP100 | 60 | 8 | 3 | 38 | 1064.27 | 495 | 434 | 95.4 |
| RP200 | 60 | 6 | 3 | 43 | 1269.02 | 629 | 698 | 93.4 |
| PCB146 | 60 | 8 | 3 | 200 | 1087.9 | 9180 | 8930 | 95.9 |
| N300 | 60 | 8 | 3 | 23 | 1185.72 | 705 | 429 | 90.3 |
| N600 | 30 | 6 | 2 | 31 | 1864.71 | 920.1 | 749 | 91.2 |
| PCB500 | 60 | 6 | 3 | 30 | 1192.1 | 1042 | 1353 | 96.3 |

- $S_i > 150$ fixam-se os centros dos itens que pertencem à solução parcial.

Nas tabelas 5.4 e 5.5 apresenta-se, para cada instância, os resultados obtidos pela heurística construtiva híbrida, indicando-se o tempo limite considerado em cada iteração (TL), o número de itens que iniciam a solução parcial (k_0), o número de itens a adicionar ao modelo nas iterações seguintes (k), o valor de Δ , o tempo total de execução, a largura e a altura do padrão, assim como a respetivo aproveitamento. Os parâmetros apresentados nestas tabelas distinguem-se essencialmente no tempo limite e no número de itens a adicionar ao modelo nas iterações diferentes da primeira. Os resultados da tabela 5.4 foram obtidos com uma configuração da heurística híbrida que privilegia a qualidade final da solução em detrimento do tempo de cálculo. Já os resultados da tabela 5.5 foram obtidos com uma configuração que privilegia a rapidez. As principais diferenças residem no tempo limite e no valor de k . Em todos os testes realizados foi considerado um *gap* igual a 0.0001 e como critério de ordenação a área dos itens por ordem decrescente.

Nas figuras 5.12 e 5.13 apresentam-se os padrões obtidos pela heurística construtiva híbrida para as instâncias retangulares, na configuração mais demorada e mais rápida, respetivamente.

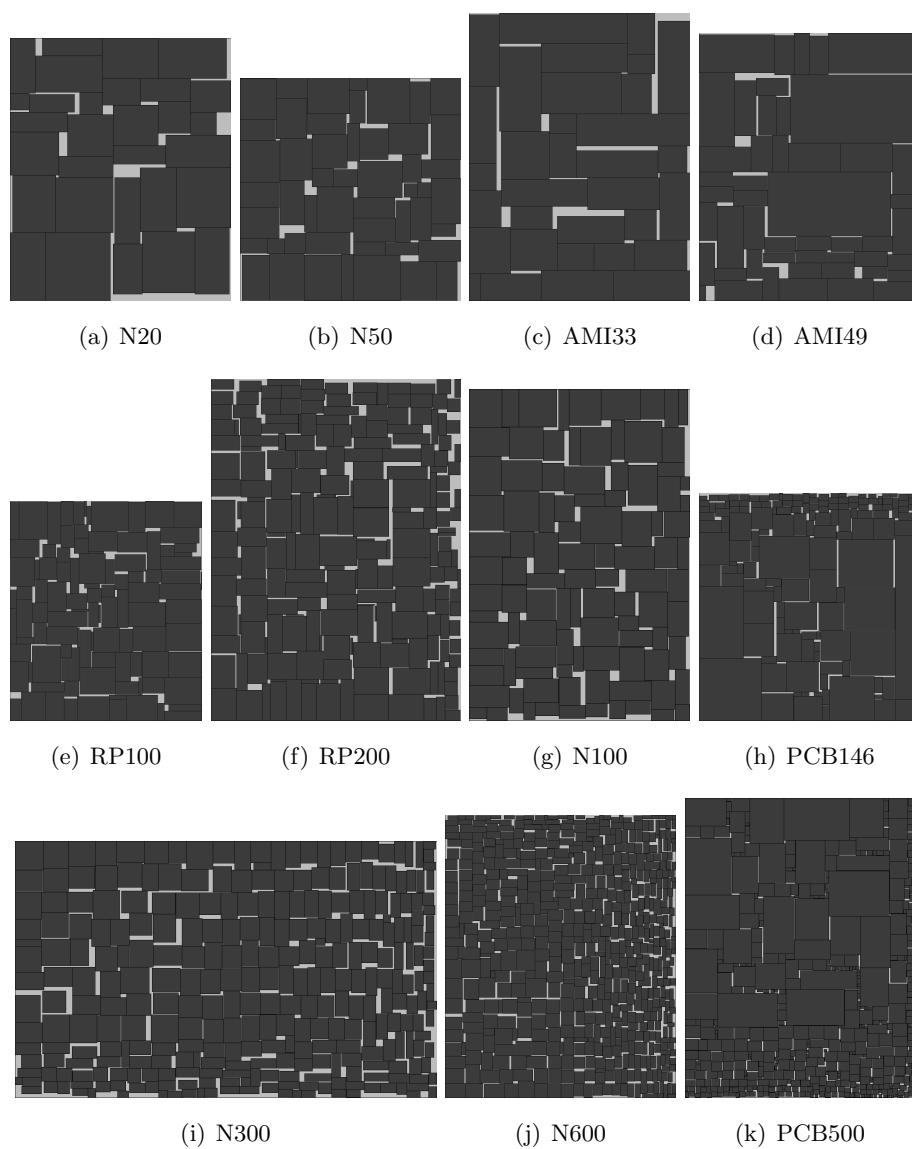


Figura 5.12: Padrões obtidos para as instâncias retangulares com a heurística construtiva híbrida numa configuração mais lenta

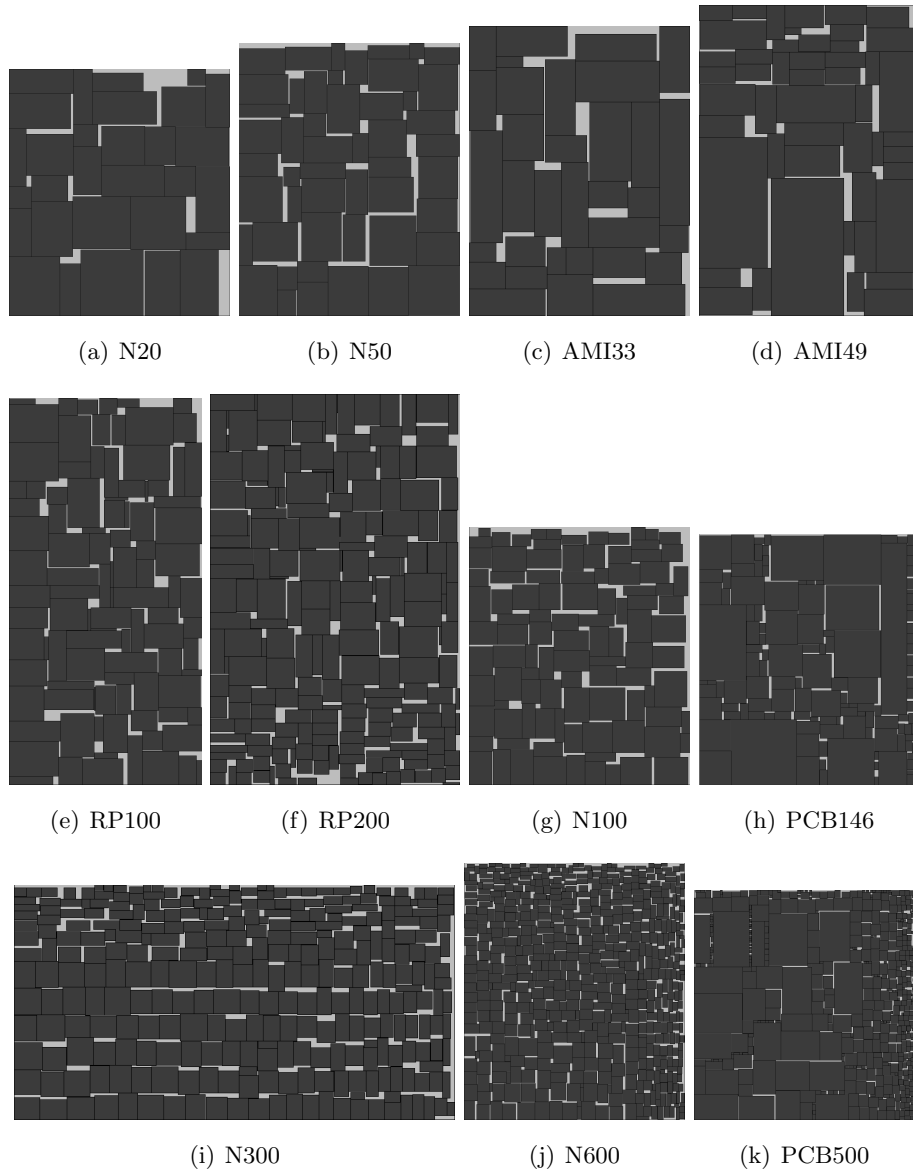


Figura 5.13: Padrões obtidos para as instâncias retangulares com a heurística construtiva híbrida numa configuração mais rápida

Tabela 5.5: Resultados dos testes computacionais da heurística construtiva híbrida numa configuração mais rápida

| Instância | TL (<i>s</i>) | k_0 | k | Δ | Tempo (<i>s</i>) | W | H | Aprov. (%) |
|-----------|-----------------|-------|-----|----------|--------------------|------|------|------------|
| N30 | 20 | 6 | 1 | 48 | 10.4264 | 501 | 447 | 93.1 |
| N50 | 20 | 6 | 1 | 41 | 21.306 | 515 | 416 | 92.7 |
| AMI33 | 20 | 6 | 1 | 91 | 10.4939 | 973 | 1281 | 92.8 |
| AMI49 | 20 | 6 | 1 | 490 | 18.0379 | 5180 | 7308 | 93.6 |
| N100 | 20 | 8 | 1 | 31 | 54.4789 | 633 | 307 | 92.4 |
| N200 | 20 | 8 | 1 | 23 | 114.697 | 309 | 623 | 88.3 |
| RP100 | 20 | 6 | 1 | 38 | 46.945 | 666 | 331 | 93.0 |
| RP200 | 20 | 6 | 1 | 43 | 111.982 | 531 | 834 | 92.6 |
| PCB146 | 20 | 8 | 1 | 200 | 68.4711 | 9630 | 8490 | 96.2 |
| N300 | 20 | 8 | 1 | 23 | 265.243 | 406 | 757 | 88.9 |
| N600 | 20 | 8 | 1 | 31 | 1511.33 | 901 | 774 | 90.1 |
| PCB500 | 20 | 6 | 1 | 30 | 559.141 | 1216 | 1168 | 95.6 |

Itens Ortogonais

As instâncias de problemas reais podem ter, para além de um conjunto maior ou menor de retângulos, um conjunto de itens ortogonais. Na tabela 5.7, apresentam-se os resultados obtidos pela heurística construtiva híbrida para duas instâncias com itens ortogonais (LIN e NAKATAKE1). A instância LIN contém vinte e nove itens dos quais 22 são retangulares e sete são ortogonais. A instância NAKATAKE1 contém quarenta itens, trinta retangulares e dez ortogonais. Os itens ortogonais foram divididos em retângulos pelo processo apresentado na secção 5.2 (figura 3.4). Na tabela 5.6 apresenta-se a identificação dos retângulos que constituem cada item ortogonal, para cada uma das instâncias. Depois desta divisão a instância LIN passou a ter quarenta retângulos no total e a instância NAKATAKE1 cinquenta.

Note-se que a instância LIN obteve 100% de aproveitamento no melhor resultado obtido na literatura. No entanto, esta instância é pouco realista visto tratar-se de um *puzzle* perfeito. Este tipo de instâncias não servem para fazer comparações, pois sabendo qual o encaixe dos itens pode construir-se uma abordagem que obtenha esse encaixe.

Na figura 5.14 apresentam-se os padrões obtidos pela heurística construtiva híbrida para as instâncias LIN e NAKATAKE1, figuras 5.14(a) e 5.14(b),

Tabela 5.6: Itens ortogonais das instâncias NAKATAKE1 e LIN

| NAKATAKE1 | | | | | | |
|----------------|----------|----|--|--|--|--|
| Item Ortogonal | Subitens | | | | | |
| 1 | 1 | 2 | | | | |
| 2 | 3 | 4 | | | | |
| 3 | 6 | 7 | | | | |
| 4 | 8 | 9 | | | | |
| 5 | 10 | 11 | | | | |
| 6 | 12 | 13 | | | | |
| 7 | 14 | 15 | | | | |
| 8 | 20 | 21 | | | | |
| 9 | 24 | 25 | | | | |
| 10 | 26 | 27 | | | | |

| LIN | | | | | | |
|----------------|----------|----|----|--|--|--|
| Item Ortogonal | Subitens | | | | | |
| 1 | 1 | 2 | 3 | | | |
| 2 | 4 | 5 | 6 | | | |
| 3 | 7 | 8 | | | | |
| 4 | 10 | 11 | 12 | | | |
| 5 | 13 | 14 | 15 | | | |
| 6 | 16 | 17 | | | | |
| 7 | 19 | 20 | | | | |

Tabela 5.7: Resultados dos testes computacionais de instâncias com itens ortogonais

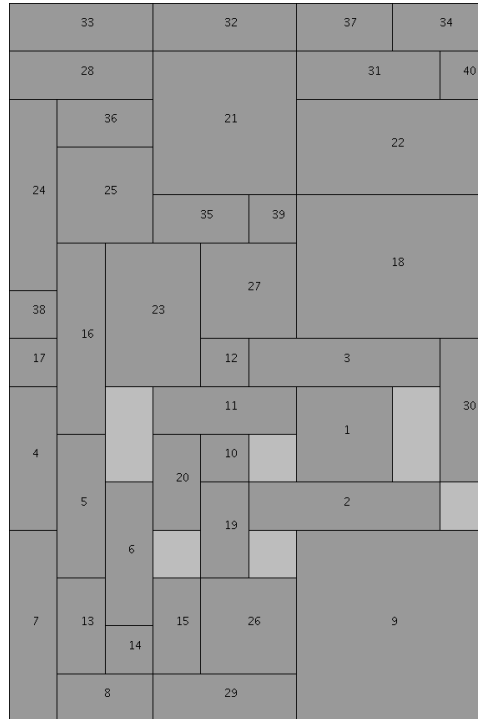
| Instância | . | | | | Tempo | | Aprov. | | Melhor Res.* | |
|-----------|--------|-------|-----|---------|-------|-----|--------|------|--------------|--|
| | # Ret. | k_0 | k | (s) | W | H | (%) | (%) | (s) | |
| LIN | 40 | 12 | 3 | 119.4 | 10 | 15 | 94.7 | 100 | 2.72 | |
| NAKATAKE1 | 50 | 11 | 2 | 177.9 | 300 | 350 | 94.1 | 96.9 | 2.72 | |
| NAKATAKE2 | 42 | 14 | 6 | 491.5 | 20 | 23 | 78.7 | 80.2 | 93.32 | |

*Chen et al. [2010]

respetivamente.

5.2.4 Comentários e Conclusões

Os testes realizados com a heurística construtiva híbrida mostram que é possível resolver instâncias de grande dimensão em tempo computacional considerado satisfatório. Uma das grandes vantagens desta heurística reside no facto de ser possível atribuir valores a alguns parâmetros que permitem que a resolução de uma dada instância seja mais ou menos rápida. Estes parâmetros são o tempo limite de resolução do modelo, em cada iteração, o *gap*, a escolha do número de itens a posicionar na primeira iteração (k_0) e o número de itens a posicionar em iterações diferentes da primeira (k). Quando se pretende obter uma boa solução estes parâmetros devem ser configurados de forma a que não seja atingido o tempo limite em cada iteração (exceto na primeira). Por exemplo, quando se aumenta o valor de k_0 deve aumentar-se o valor do tempo limite. Em relação a instâncias com itens ortogonais, os resultados obtidos com a heurística construtiva híbrida foram superiores aos obtidos pela heurística Rect-TOPOS apresentada na secção 4.2. Deve notar-se que a abordagem usada por [Bortfeldt, 2013] lida com combinações lineares entre as dimensões dos itens. Assim, por exemplo, a instância RP200, apesar de ter mais retângulos do que a instância PCB146, o tempo de execução da primeira é muito menor do que o da segunda, porque é mais fácil combinar dimensões entre os itens. Apesar das instâncias com itens ortogonais, existentes na literatura, serem de média dimensão é possível com a heurística construtiva híbrida resolver instâncias de grande dimensão.



(a) LIN



(b) NAKATAKE1

Figura 5.14: Padrões obtidos para as instâncias LIN e NAKATAKE1

Capítulo 6

Conclusão

6.1 Conclusões do Trabalho

Nesta dissertação considerou-se o PP de itens ortogonais num objeto onde as dimensões não estão definidas. O principal objetivo desta tese foi desenvolver modelos matemáticos e abordagens heurísticas para lidar com instâncias de grande dimensão. Como resposta à primeira questão de investigação colocada no início deste trabalho que consistia em decidir qual a abordagem a usar para o problema proposto pode afirmar-se que, através das experiências realizadas, tal como já era esperado, não é possível resolver, com o modelo global, instâncias de média e grande dimensão até à otimalidade. Assim, foi necessário usar abordagens baseadas em métodos heurísticos para a resolução do PP, para instâncias de média/grande dimensão, num tempo computacional reduzido. Foram propostos métodos construtivos e métodos híbridos (que combinam modelos matemáticos com métodos heurísticos). Quando foi necessário decidir que regra usar na ordenação dos itens, para posteriormente se proceder ao seu posicionamento, optou-se por posicionar, em primeiro lugar, os itens maiores (maior área, maior perímetro, maior dimensão). Esta decisão baseou-se num estudo, recorrendo a uma heurística construtiva parametrizável.

Foi proposta uma adaptação de uma heurística construtiva que opera diretamente no padrão, que foi originalmente desenvolvida para lidar com itens irregulares. Trata-se de uma heurística rápida que lida facilmente com todo o tipo de polígonos ortogonais (Rect-TOPOS). A qualidade das soluções obtidas revelou-se bastante satisfatória dado tratar-se de uma simples heurística com reduzidos tempos de execução.

Tentando tirar partido das vantagens existentes nos modelos matemáticos e nos métodos heurísticos foram propostas duas abordagens que combinam estes dois métodos de resolução. Uma das abordagens combina a meta-heurística *Iterated Local Search* (ILS) com modelos matemáticos de compactação e separação. A utilização destes dois modelos de forma coordenada permitiu a obtenção de padrões compactos e admissíveis a partir de um padrão inicial que poderia ser não admissível. Uma segunda abordagem utilizou modelos matemáticos na construção iterativa de soluções onde em cada iteração um conjunto de itens é adicionado a uma solução parcial, por intermédio de um modelo matemático. A escolha dos itens a posicionar em cada iteração foi feita segundo critérios heurísticos.

Em relação à questão de investigação colocada acerca da forma mais indicada de lidar com itens ortogonais, isto é, se devem ser divididos em retângulos ou não, em algumas das abordagens propostas nesta dissertação optou-se pela divisão dos itens ortogonais em retângulos e noutras optou-se por lidar com o item como um todo.

Uma importante contribuição desta dissertação foi a extensão do modelo, existente para itens retangulares, para lidar com itens ortogonais. Para este modelo foi realizado um conjunto de experiências para validar a extensão do modelo a itens ortogonais. Este modelo (que lida com itens ortogonais) foi também incorporado na heurística construtiva híbrida para lidar com instâncias com itens ortogonais.

6.2 Desenvolvimentos Futuros

Nesta secção apresentam-se algumas perspectivas de desenvolvimento futuro deste trabalho.

Os testes realizados com a heurística Rect-TOPOS, sugerem a incorporação desta heurística numa abordagem baseada em pesquisa local. Nesta abordagem seria possível, à custa de um aumento do tempo de execução, melhorar os já bons resultados alcançados pelo Rect-TOPOS em situações onde tal fosse necessário.

O tempo de execução para o cálculo do invólucro de posicionamento entre pares de itens sugere o desenvolvimento de um método específico para lidar com itens ortogonais.

Uma outra possibilidade de investigação seria desenvolver uma abor-

dagem híbrida de pesquisa local com base no mesmo modelo da heurística híbrida baseada na construção de padrões. O objetivo desta abordagem seria melhorar iterativamente soluções existentes através de um processo de pesquisa local que consistiria na manipulação (fixação e relaxação) de variáveis binárias respeitantes às posições relativas entre pares de itens. Um conjunto de itens seria selecionado para ser removido do padrão e, conseqüentemente, todas as variáveis binárias referentes a esses itens seriam libertadas. Os itens selecionados seriam novamente posicionados, na sua totalidade, numa única iteração ou, poderiam ser posicionados em várias iterações, recorrendo à heurística construtiva híbrida.

Finalmente, uma linha de investigação diferente seria o desenvolvimento de abordagens para PP tridimensionais com itens ortogonais.

Bibliografia

Adya, S. e Markov, I., 2003. Fixed-outline floorplanning: enabling hierarchical design. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 11(6):1120–1135.

Alvarez-Valdes, R., Parreño, F., e Tamarit, J.M., 2005. A tabu search algorithm for the pallet loading problem. *OR Spectrum*, 27(1):43–61.

Alvarez-Valdes, R., Parreño, F., e Tamarit, J.M., 2008. A branch and bound algorithm for the strip packing problem. *OR Spectrum*, 31(2):431–459.

Art, R., 1966. An approach to the two dimensional irregular cutting stock problem. Relatório Técnico 36. Y08, IBM Cambridge Scientific Center, Cambridge, Massachusetts, USA.

Baker, B.S., Coffman, Jr., E.G., e Rivest, R.L., 1980. Orthogonal packings in two dimensions. *SIAM Journal on Computing*, 9(4):846–855.

Beasley, J.E., 1985. An exact two-dimensional non-guillotine cutting tree search procedure. *Operations Research*, 33(1):49–64.

Beltrán-Cano, D., 2009. Solving the rectangle packing problem by an iterative hybrid heuristic. In *Computer Aided Systems Theory – EUROCAST 2009*, capítulo Solving th, páginas 673–680. Springer Berlin / Heidelberg.

Bennell, J.A. e Dowsland, K.A., 2001. Hybridising tabu search with optimisation techniques for irregular stock cutting. *Management Science*, 47(8):1160–1172.

Berkey, J. e Wang, P., 1987. Two-dimensional finite bin-packing algorithms. *Journal of the operational research society*, 38(5):423–429.

- Birgin, E., Martínez, J., e Ronconi, D., 2005a. Optimizing the packing of cylinders into a rectangular container: A nonlinear approach. *European Journal of Operational Research*, 160(1):19–33.
- Birgin, E.G., Morabito, R., e Nishihara, F.H., 2005b. A note on an L-approach for solving the manufacturer's pallet loading problem. *Journal of the Operational Research Society*, 56(12):1448–1451.
- Bischoff, E.E. e Marriott, M.D., 1990. A comparative evaluation of heuristics for container loading. *European Journal of Operational Research*, 44(2):267–276.
- Bortfeldt, A., 2006. A genetic algorithm for the two-dimensional strip packing problem with rectangular pieces. *European Journal of Operational Research*, 172(3):814–837.
- Bortfeldt, A., 2013. A reduction approach for solving the rectangle packing area minimization problem. *European Journal of Operational Research*, 224(3):486–496.
- Bortfeldt, A. e Gehring, H., 2001. A hybrid genetic algorithm for the container loading problem. *European Journal of Operational Research*, 131(1):143–161.
- Caldwell, A., Kahng, A., Mantik, S., Markov, I., e Zelikovsky, A., 1999. On wirelength estimations for row-based placement. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 18(9):1265–1278.
- Cemil Dikili, A., 2004. A new approach for the solution of the two-dimensional guillotine-cutting problem in ship production. *Ocean Engineering*, 31(8-9):1193–1203.
- Chan, H.H. e Markov, I.L., 2004. Practical slicing and non-slicing block-packing without simulated annealing. In *Proceedings of the 14th ACM Great Lakes symposium on VLSI - GLSVLSI '04*, página 282. ACM Press, New York, New York, USA.
- Chang, Y.C., Chang, Y.W., Wu, G.M., e Wu, S.W., 2000. B*-trees. In *Proceedings of the 37th conference on Design automation - DAC '00*, páginas 458–463. ACM Press, New York, New York, USA.

- Chatterjee, D. e Manikas, T., 2007. A genetic algorithm for non-slicing floorplan representation. *National Conference on Intelligent Systems*.
- Chazelle, B., 1983. The bottom-left bin-packing heuristic: An efficient implementation. *IEEE Transactions on Computers*, C-32(8):697–707.
- Chen, C., Lee, S., e Shen, Q., 1995. An analytical model for the container loading problem. *European Journal of Operational Research*, 80(1):68–76.
- Chen, C.S., Sarin, S., e Ram, B., 1993. A mixed-integer programming model for a class of assortment problems. *European Journal of Operational Research*, 65(3):362–367.
- Chen, D., Liu, J., Fu, Y., e Shang, M., 2010. An efficient heuristic algorithm for arbitrary shaped rectilinear block packing problem. *Computers & Operations Research*, 37(6):1068–1074.
- Chen, G., Guo, W., e Chen, Y., 2009. A PSO-based intelligent decision algorithm for VLSI floorplanning. *Soft Computing*, 14(12):1329–1337.
- Chen, P. e Kuh, E.S., 2000. Floorplan sizing by linear programming approximation. In *Proceedings of the 37th conference on Design automation - DAC '00*, páginas 468–471. ACM Press, New York, New York, USA.
- Chen, T. e Fan, M.K.H., 1998. On convex formulation of the floorplan area minimization problem. In *Proceedings of the 1998 international symposium on Physical design - ISPD '98*, ISPD '98, páginas 124–128. ACM Press, New York, New York, USA.
- Chen, X., Hu, J., e Xu, N., 2011. Regularity-constrained floorplanning for multi-core processors. In *Proceedings of the 2011 international symposium on Physical design - ISPD '11*, página 99. ACM Press, New York, New York, USA.
- Christofides, N. e Hadjiconstantinou, E., 1995. An exact algorithm for orthogonal 2-D cutting problems using guillotine cuts. *European Journal of Operational Research*, 83(1):21–38.
- Christofides, N. e Whitlock, C., 1977. An algorithm for two-dimensional cutting problems. *Operations Research*, 25(1):30–44.

- Chu, C. e Young, E., 2004. Nonrectangular shaping and sizing of soft modules for floorplan-design improvement. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 23(1):71–79.
- Clímaco, J.N., Antunes, C.H., e Alves, M.J., 2003. *Programação Linear Multiobjetivo*. Imprensa da Universidade de Coimbra.
- Coelho, D.G., Wanner, E.F., Souza, S.R., Carrano, E.G., e Purshouse, R.C., 2012. A multiobjective evolutionary algorithm for the 2D guillotine strip packing problem. In *2012 IEEE Congress on Evolutionary Computation*, páginas 1–8. IEEE.
- Correia, M.H., Oliveira, J.F., e Ferreira, J.S., 2000. Cylinder packing by simulated annealing. *Pesquisa Operacional*, 20(2):269–286.
- Costa, M.T., Gomes, A.M., e Oliveira, J.F., 2009. Heuristic approaches to large-scale periodic packing of irregular shapes on a rectangular sheet. *European Journal of Operational Research*, 192(1):29–40.
- Crispin, A.J., Clay, P., Taylor, G.E., Bayes, T., e Reedman, D., 2003. Genetic algorithms applied to leather lay plan material utilization. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 217(12):1753–1756.
- Cui, Y., 2005. Generating optimal T-shape cutting patterns for circular blanks. *Computers & Operations Research*, 32(1):143–152.
- Dong, S., Yang, Z., Hong, X., e Wu, Y., 2004. Module placement based on quadratic programming and rectangle packing using less flexibility first principle. In *2004 IEEE International Symposium on Circuits and Systems (IEEE Cat. No.04CH37512)*, páginas V–61–V–64. IEEE.
- Dowsland, K.A., 1987. An exact algorithm for the pallet loading problem. *European Journal of Operational Research*, 31(1):78–84.
- Drakidis, A., Mack, R., e Massara, R., 2006. Packing-based VLSI module placement using genetic algorithm with sequence-pair representation. In *IEE Proceedings - Circuits, Devices and Systems*, 153, páginas 545–551.
- Dumitrescu, I. e Stützle, T., 2003. Combinations of local search and exact algorithms. In *Applications of Evolutionary Computing*, volume 2611 de

Lecture Notes in Computer Science, páginas 57–68. Springer Berlin / Heidelberg.

Dyckhoff, H., 1990. A typology of cutting and packing problems. *European Journal of Operational Research*, 44(2):145–159.

Faina, L., 1999. An application of simulated annealing to the cutting stock problem. *European Journal of Operational Research*, 114(3):542–556.

Fang, J. e Wong, J., 1991. A fast constraint graph based compactor for VLSI circuit layouts. In *1991 International Conference on Circuits and Systems*, volume 2, páginas 628–631. IEEE.

Fekete, S.P., Schepers, J., e van der Veen, J., 2007. An exact algorithm for higher-dimensional orthogonal packing. *Operations Research*, 55(3):569–587.

Feltl, H. e Raidl, G.R., 2004. An improved hybrid genetic algorithm for the generalized assignment problem. In *Proceedings of the 2004 ACM symposium on Applied computing - SAC '04*, SAC '04, página 990. ACM Press, New York, New York, USA.

Fernando, P. e Katkoori, S., 2008. An elitist non-dominated sorting based genetic algorithm for simultaneous area and wirelength minimization in VLSI floorplanning. *21st International Conference on VLSI Design (VLSI-SID 2008)*, páginas 337–342.

Ferreira, E.P. e Oliveira, J.F., 2008. Fekete and Schepers' graph-based algorithm for the two-dimensional orthogonal packing problem revisited. In A. Bortfeldt, J. Homberger, H. Kopfer, G. Pankratz, e R. Strangmeier, editores, *Intelligent Decision Support*, páginas 15–31. Gabler.

Fischetti, M. e Lodi, A., 2003. Local branching. *Mathematical Programming*, 98(1-3):23–47.

Fujiyoshi, K. e Murata, H., 2000. Arbitrary convex and concave rectilinear block packing using sequence-pair. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 19(2):224–233.

Fujiyoshi, K. e Murata, H., 2002. Packing of L-shaped and rectangular modules using sequence-pair method. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, 85(2):52–64.

- G, Y.G., Seong, Y.J., e Kang, M.K., 2003. A best-first branch and bound algorithm for unconstrained two-dimensional cutting problems. *Operations Research Letters*, 31(4):301–307.
- Garey, M.R. e Johnson, D.S., 1979. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman & Co., New York, USA.
- Geiger, M.J., 2008. Bin packing under multiple objectives - a heuristic approximation approach. *arXiv preprint arXiv:0809.0755*, páginas 53–56.
- George, J.A., George, J.M., e Lamar, B.W., 1995. Packing different-sized circles into a rectangular container. *European Journal of Operational Research*, 84(3):693–712.
- Gilmore, P.C. e Gomory, R.E., 1961. A linear programming approach to the cutting-stock problem. *Operations Research*, 9(6):849–859.
- Gilmore, P.C. e Gomory, R.E., 1963. A linear programming approach to the cutting stock problem—Part II. *Operations Research*, 11(6):863–888.
- Gilmore, P.C. e Gomory, R.E., 1965. Multistage cutting stock problems of two and more dimensions. *Operations Research*, 13(1):94–120.
- Gilmore, P.C. e Gomory, R.E., 1966. The theory and computation of knapsack functions. *Operations Research*, 14(6):1045–1074.
- Gomes, A.M. e Oliveira, J.F., 2006. Solving irregular strip packing problems by hybridising simulated annealing and linear programming. *European Journal of Operational Research*, 171(3):811–829.
- Gradišar, M., Kljajić, M., Resinovič, G., e Jesenko, J., 1999. A sequential heuristic procedure for one-dimensional cutting. *European Journal of Operational Research*, 114(3):557–568.
- Hadjiconstantinou, E. e Christofides, N., 1995. An exact algorithm for general, orthogonal, two-dimensional knapsack problems. *European Journal of Operational Research*, 83(1):39–56.
- Hakimi, S., 1996. Globally optimal floorplanning for a layout problem. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 43(9):713–720.

- Heistermann, J. e Lengauer, T., 1995. The nesting problem in the leather manufacturing industry. *Annals of Operations Research*, 57(1):147–173.
- Herz, J.C., 1972. Recursive computational procedure for two-dimensional stock cutting. *IBM Journal of Research and Development*, 16(5):462–469.
- Hifi, M., 1998. Exact algorithms for the guillotine strip cutting/packing problem. *Computers & Operations Research*, 25(11):925–940.
- Hopper, E. e Turton, B., 2001. An empirical investigation of meta-heuristic and heuristic algorithms for a 2D packing problem. *European Journal of Operational Research*, 128(1):34–57.
- Ibaraki, T., Imahori, S., e Yagiura, M., 2008. Hybrid metaheuristics for packing problems. In *Hybrid Metaheuristics*, volume 114, páginas 185–219. Springer Berlin / Heidelberg.
- Imahori, S., Kushiya, M., Nakashima, T., e Sugihara, K., 2008. Generation of cutter paths for hard material in wire EDM. *Journal of Materials Processing Technology*, 206(1-3):453–461.
- Imahori, S., Yagiura, M., e Ibaraki, T., 2003. Local search algorithms for the rectangle packing problem with general spatial costs. *Mathematical Programming*, 97(3):543–569.
- Imahori, S., Yagiura, M., e Ibaraki, T., 2005. Improved local search algorithms for the rectangle packing problem with general spatial costs. *European Journal of Operational Research*, 167(1):48–67.
- Iori, M., Martello, S., e Monaci, M., 2003. Metaheuristic algorithms for the strip packing problem. In *Optimization and Industry: New Frontiers*. Springer.
- Jagannathan, A., Hur, S.W., e Lillis, J., 2002. A fast algorithm for context-aware buffer insertion. *ACM Transactions on Design Automation of Electronic Systems*, 7(1):173–188.
- Jourdan, L., Basseur, M., e Talbi, E.G., 2009. Hybridizing exact methods and metaheuristics: A taxonomy. *European Journal of Operational Research*, 199(3):620–629.

- Kang, M. e Dai, W.M., 1998a. Arbitrary rectilinear block packing based on sequence pair. In *Computer-Aided Design, 1998. ICCAD 98. Digest of Technical Papers. 1998 IEEE/ACM International Conference on*, páginas 259–266.
- Kang, M.Z.W. e Dai, W.W.M., 1998b. Topology constrained rectilinear block packing for layout reuse. In *Proceedings of the 1998 international symposium on Physical design - ISPD '98, ISPD '98*, páginas 179–186. ACM Press, New York, New York, USA.
- Kantarovich, L.V., 1960. A mathematical methods of organizing and planning production. *Management Science*, 6(4):366–422.
- Kato, N. e Ibaraki, T., 1987. A parametric characterization and an ϵ -approximation scheme for the minimization of a quasiconcave program. *Discrete Applied Mathematics*, 17(1-2):39–66.
- Kim, J. e Kim, Y.D., 2003. A linear programming-based algorithm for floorplanning in VLSI design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 22(5):584–592.
- Korf, R.E., Moffitt, M.D., e Pollack, M.E., 2008. Optimal rectangle packing. *Annals of Operations Research*, 179(1):261–295.
- Kusiak, A. e Heragu, S.S., 1987. The facility layout problem. *European Journal of Operational Research*, 29(3):229–251.
- Lee, T.C., 1993. A bounded 2D contour searching algorithm for floorplan design with arbitrarily shaped rectilinear and soft modules. In *Design Automation, 1993. 30th Conference on*, páginas 525–530.
- Lesh, N., Marks, J., McMahon, A., e Mitzenmacher, M., 2004. Exhaustive approaches to 2D rectangular perfect packings. *Information Processing Letters*, 90(1):7–14.
- Li, H.L. e Chang, C.T., 1998. An approximately global optimization method for assortment problems. *European Journal of Operational Research*, 105(3):604–612.
- Li, H.L., Chang, C.T., e Tsai, J.F., 2002. Approximately global optimization for assortment problems using piecewise linearization techniques. *European Journal of Operational Research*, 140(3):584–589.

- Li, H.L. e Tsai, J.F., 2001. A fast algorithm for assortment optimization problems. *Computers & Operations Research*, 28(12):1245–1252.
- Li, Y., Li, Y., e Zhou, M., 2010. Area optimization in floorplanning using AP-TCG. *Journal of Convergence Information Technology*, 5(10):216–222.
- Li, Y., Li, Y., e Zhou, M., 2011. A greedy algorithm for wire length optimization. In *2011 18th IEEE International Conference on Electronics, Circuits, and Systems*, páginas 366–369. IEEE.
- Lim, S., 2008. *Practical problems in VLSI physical design automation*. Springer-Verlag, New York.
- Lin, C.C., 2006. A genetic algorithm for solving the two-dimensional assortment problem. *Computers & Industrial Engineering*, 50(1-2):175–184.
- Lin, J., Chen, H., e Chang, Y., 2002. Arbitrarily shaped rectilinear module placement using the transitive closure graph representation. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 10(6):886–901.
- Lin, J.M. e Chang, Y.W., 2001. TCG: a transitive closure graph-based representation for non-slicing floorplans. In *Proceedings Design Automation Conference, 2001*, páginas 764–769.
- Lins, L., Lins, S., e Morabito, R., 2003. An L-approach for packing (l, w)-rectangles into rectangular and L-shaped pieces. *Journal of the Operational Research Society*, 54(7):777–789.
- Liu, D. e Teng, H., 1999. An improved BL-algorithm for genetic algorithm of the orthogonal packing of rectangles. *European Journal of Operational Research*, 112(2):413–420.
- Lodi, A., Martello, S., e Vigo, D., 2002. Recent advances on two-dimensional bin packing problems. *Discrete Applied Mathematics*, 123(1-3):379–396.
- Lourenço, H.R., Martin, O.C., e Stützle, T., 2003. Iterated local search. In F. Glover e G. Kochenberger, editores, *Handbook of Metaheuristics*, volume 57 de *International Series in Operations Research & Management Science*, páginas 320–353. Springer US.

- Martello, S., Monaci, M., e Vigo, D., 2003. An exact approach to the strip-packing problem. *INFORMS Journal on Computing*, 15(3):310–319.
- Martello, S. e Toth, P., 1990. Knapsack problems: algorithms and computer implementations. *John Wiley & Sons*.
- Martello, S. e Vigo, D., 1998. Exact solution of the two-dimensional finite bin packing problem. *Management Science*, 44(3):388–399.
- Morabito, R. e Garcia, V., 1998. The cutting stock problem in a hardboard industry: A case study. *Computers & Operations Research*, 25(6):469–485.
- Morabito, R. e Morales, S., 1998. A simple and effective recursive procedure for the manufacturer's pallet loading problem. *Journal of the Operational Research Society*, 49:819–828.
- Moreira, L.M., 2001. Determinação de percursos de corte numa fábrica de ferramentas de alta precisão. *Tese de Mestrado, Escola de Gestão do Porto, Universidade do Porto*.
- Moreira, L.M., Oliveira, J.F., Gomes, A.M., e Ferreira, J.S., 2007. Heuristics for a dynamic rural postman problem. *Computers & Operations Research*, 34(11):3281–3294.
- Mornar, V. e Khoshnevis, B., 1997. A cutting stock procedure for printed circuit board production. *Computers & Industrial Engineering*, 32(1):57–66.
- Moura, A. e Oliveira, J.F., 2008. An integrated approach to the vehicle routing and container loading problems. *OR Spectrum*, 31(4):775–800.
- Murata, H., Fujiyoshi, K., Nakatake, S., e Kajitani, Y., 1995. Rectangle-packing-based module placement. In *Proceedings of IEEE International Conference on Computer Aided Design (ICCAD)*, ICCAD '95, páginas 472–479. IEEE Comput. Soc. Press.
- Nakatake, S. e Fujiyoshi, K., 1997. Module placement on BSG-structure and IC layout applications. In *ICCAD '96 Proceedings of the 1996 IEEE/ACM international conference on Computer-aided design*, ICCAD '96, páginas 484–491. IEEE Computer Society.

Nakatake, S., Furuya, M., e Kajitani, Y., 1998. Module placement on BSG-structure with pre-placed modules and rectilinear modules. In *Proceedings of 1998 Asia and South Pacific Design Automation Conference*, páginas 571–576. IEEE.

Okano, H., 2002. A scanline-based algorithm for the 2D free-form bin packing problem. *Journal of the Operations Research Society of Japan*, 45(2):145–161.

Oliveira, J. e Ferreira, J., 1990. An improved version of Wang’s algorithm for two-dimensional cutting problems. *European Journal of Operational Research*, 44(2):256–266.

Oliveira, J.F., Costa, T., e Wäscher, G., 2011a. Testes computacionais em problemas de cortes e empacotamentos bidimensionais. *IO 2011 - 15º Congresso da APDIO*.

Oliveira, J.F., Gomes, A.M., e Ferreira, J.S., 2000. TOPOS – a new constructive algorithm for nesting problems. *OR Spektrum*, 22(2):263.

Oliveira, M., Pinto Ferreira, E., e Gomes, A.M., 2009. A hybrid approach to the rectangle packing area minimization problem. In *EU/ME European Chapter on Metaheuristics*, páginas 147–152.

Oliveira, M., Pinto Ferreira, E., e Gomes, A.M., 2011b. Estudo paramétrico de uma heurística construtiva configurável para o problema da minimização da Área da envolvente rectangular no posicionamento de rectângulos. In *Anais do XLIII SBPO*, páginas 2083–2104.

Oliveira, M., Pinto Ferreira, E., e Gomes, A.M., 2011c. Rect-TOPOS: A constructive heuristic for the rectilinear packing area minimization problem. In *Proceedings of the VII ALIO?EURO – Workshop on Applied Combinatorial Optimization*, páginas 66–69.

Oliveira, M., Pinto Ferreira, E., e Gomes, A.M., 2011d. Rect-TOPOS: Uma heurística construtiva para a minimização da Área da envolvente rectangular no posicionamento de formas com componentes rectangulares. In *Livro de Actas do 15º Congresso da Associação Portuguesa de Investigação Operacional IO 2011*, páginas 165–176.

- Onodera, H., Taniguchi, Y., e Tamaru, K., 1991. Branch-and-bound placement for building block layout. In *Proceedings of the 28th conference on ACM/IEEE design automation conference - DAC '91*, páginas 433–439. ACM Press, New York, New York, USA.
- Pang, Y., Cheng, C.K., Lampaert, K., e Xie, W., 2001. Rectilinear block packing using O-tree representation. In *Proceedings of the 2001 international symposium on Physical design - ISPD '01*, páginas 156–161. ACM Press, New York, New York, USA.
- Pentico, D.W., 2008. The assortment problem: A survey. *European Journal of Operational Research*, 190(2):295–309.
- Pisinger, D., 1995. An expanding-core algorithm for the exact 0–1 knapsack problem. *European Journal of Operational Research*, 87(1):175–187.
- Pisinger, D., 1999. An exact algorithm for large multiple knapsack problems. *European Journal of Operational Research*, 114(3):528–541.
- Pisinger, D. e Sigurd, M., 2005. The two-dimensional bin packing problem with variable bin sizes and costs. *Discrete Optimization*, 2(2):154–167.
- Plateau, A., Tachet, D., e Tolla, P., 2002. A hybrid search combining interior point methods and metaheuristics for 0-1 programming. *International Transactions in Operational Research*, 9(6):731–746.
- Preas, B. e VanCleemput, W., 1979. Placement algorithms for arbitrarily shaped blocks. In *16th Design Automation Conference, 25 years of DAC*, páginas 474–480. IEEE.
- Puchinger, J. e Raidl, G., 2005. Combining metaheuristics and exact algorithms in combinatorial optimization: A survey and classification. In *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, volume 3562, páginas 113–124. Springer Berlin / Heidelberg.
- Raidl, G., 2006. A unified view on hybrid metaheuristics. In *Hybrid Metaheuristics*, volume 4030 de *Lecture Notes in Computer Science*, páginas 1–12. Springer Berlin / Heidelberg.
- Raidl, G. e Kodydek, G., 1998. Genetic algorithms for the multiple container packing problem. In *Parallel Problem Solving from Nature – PPSN V*, páginas 875–884. Springer-Verlag.

- Raidl, G. e Puchinger, J., 2008. Combining (integer) linear programming techniques and metaheuristics for combinatorial optimization. In *Hybrid Metaheuristics*, páginas 31–62. Springer Berlin / Heidelberg.
- Ripon, K.S.N., Glette, K., Høvin, M., e Tørresen, J., 2010. Multi-objective evolutionary approach for solving facility layout problem using local search. In *Proceedings of the 2010 ACM Symposium on Applied Computing - SAC '10*, SAC '10, página 1155. ACM Press, New York, New York, USA.
- Sathe, M., Schenk, O., e Burkhart, H., 2009. Solving bi-objective many-constraint bin packing problems in automobile sheet metal forming processes. In *Evolutionary Multi-Criterion Optimization*, volume 5467, páginas 246–260. Springer Berlin / Heidelberg.
- Sharma, R., Balachander, T., McCord, C., An, S., e Zhang, Q., 1997. Genetic algorithm for the non-convex cutting stock problem. *Transactions of the North American Manufacturing Research Institute*, XXV:281–286.
- Singh, S. e Singh, V., 2010. An improved heuristic approach for multi-objective facility layout problem. *International Journal of Production Research*, 48(4):1171–1194.
- Sutanthavibul, S., Shragowitz, E., e Rosen, J., 1991. An analytical approach to floorplan design and optimization. In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, volume 10, páginas 761–769.
- Talbi, E.G., 2002. A taxonomy of hybrid metaheuristics. *Journal of heuristics*, 8(5):541–564.
- Tang, X., Tian, R., e Wong, M., 2006. Minimizing wire length in floorplaning. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(9):1744–1753.
- Tang, X. e Wong, M., 2004. On handling arbitrary rectilinear shape constraint. In *Proceedings of the 2004 Asia and South Pacific Design Automation Conference*, páginas 38–41.
- Tay, F.E., Chong, T., e Lee, F., 2002. Pattern nesting on irregular-shaped stock using genetic algorithms. *Engineering Applications of Artificial Intelligence*, 15(6):551–558.

- van Stee, R., 2004. An approximation algorithm for square packing. *Operations Research Letters*, 32(6):535–539.
- Vanderbeck, F., 2001. A nested decomposition approach to a three-stage, two-dimensional cutting-stock problem. *Management Science*, 47(6):864–879.
- Wang, P.Y., 1983. Two algorithms for constrained two-dimensional cutting stock problems. *Operations Research*, 31(3):573–586.
- Wäscher, G., Haußner, H., e Schumann, H., 2007. An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3):1109–1130.
- Watanabe, S. e Hiroyasu, T., 2004. Multi-objective rectangular packing problem and its applications. In *Evolutionary Multi-Criterion Optimization*, páginas 581–603.
- Wei, L., Zhang, D., e Chen, Q., 2009. A least wasted first heuristic algorithm for the rectangular packing problem. *Computers & Operations Research*, 36(5):1608–1614.
- Wong, D. e Liu, C., 1986. A new algorithm for floorplan design. In *DAC '86 Proceedings of the 23rd ACM/IEEE Design Automation Conference*, DAC '86, páginas 101–107.
- Wu, G.M., Chang, Y., e Chang, Y.W., 2000. Rectilinear block placement using B*-trees. In *Proceedings 2000 International Conference on Computer Design*, páginas 351–356. IEEE Comput. Soc.
- Xu, J., Guo, P.n., e Cheng, C.K., 1998. Rectilinear block placement using sequence-pair. In *Proceedings of the 1998 international symposium on Physical design - ISPD '98*, ISPD '98, páginas 173–178. ACM Press, New York, New York, USA.
- Yamazaki, H., Sakanushi, K., e Kajitani, Y., 2000. Optimum packing of convex-polygons by a new data structure sequence-table. In *IEEE APC-CAS 2000. 2000 IEEE Asia-Pacific Conference on Circuits and Systems. Electronic Communication Systems*, páginas 821–824. IEEE.

Ye, M. e Zhou, G., 2007. A local genetic approach to multi-objective, facility layout problems with fixed aisles. *International Journal of Production Research*, 45(22):5243–5264.

Yoshimura, T., 1999. An O-tree representation of non-slicing floorplan and its applications. In *Proceedings 1999 Design Automation Conference (Cat. No. 99CH36361)*, DAC '99, páginas 268–273. IEEE.

Zhao, C., Chen, J., Zhou, X., Jing, M., e Zhou, D., 2006. Floorplanning method based on linear programming. In *Proceedings of the 10th WSEAS international conference on Systems*, ICS'06, páginas 623–629.